Smart Antenna Design for tracking multiple users using FPGA virtex-5

Manoj Govind Chaudhari

Department of Electrical Engineering
National Institute of Technology,Rourkela
Rourkela-769008, Odisha, INDIA
May 2015

Smart Antenna Design for tracking multiple users using FPGA virtex-5

A thesis submitted in partial fulfillment of the
requirements for the degree of

Master of Technology

in

Electrical Engineering

by

Manoj Govind Chaudhari
(Roll-213EE1281)

Under the Guidance of

Prof.K. R. Subhashini

Department of Electrical Engineering
National Institute of Technology,Rourkela
Rourkela-769008, Odisha, INDIA
2013-2015

Department of Electrical Engineering

National Institute of Technology, Rourkela

C E R T I F I C A T E

This is to certify that the thesis entitled ***"Smart Antenna Design for tracking multiple users using FPGA virtex-5"*** *by* ***Mr. Manoj Govind Chaudhari****, submitted to the National Institute of Technology, Rourkela (Deemed University) for the award of Master of Technology in Electrical Engineering, is a record of bonafide research work carried out by him in the Department of Electrical Engineering , under my supervision. I believe that this thesis fulfills the requirements for the award of degree of Master of Technology.The results embodied in the thesis have not been submitted for the award of any other degree elsewhere.*

Prof.K. R. Subhashini

Place:Rourkela

Date:

To My parents, friends and GUIDE

Acknowledgements

I am thankful to my supervisor Professor K. R. Subhashini for her inspiration, excellent guidance and confidence through my study, without which this thesis would not be in its present form. I also thank her for all the encouragement throughout this project work.

I express my gratitude to the members of Masters Scrutiny Committee, "Professors D. Patra, S. Das, P. K. Sahoo, Supratim Gupta" for their advise. I am also very much obliged to Head of the Department of Electrical Engineering, NIT Rourkela for providing all the required facilities towards this work. I also thank other faculty members in the department for their invaluable support.

I would like to thank my colleagues for their enjoyable and helpful company I had with them.

My wholehearted gratitude to my parents, for their encouragement and support.

Manoj Govind Chaudhari
Rourkela, May 2015

Contents

Abstract

Smart Antenna systems have received increasing interest recently as the demands for good quality and latest value added services on the available wireless communication systems are increasing. There is an increase in demand on mobile wireless operators to provide voice and high-speed data services and to support more users in a single base station in order to reduce costs for overall network and make the services affordable to subscribers. Smart antenna technology offers a significant improved solution to reduce interference levels and improve the capacity of the system. Although lot of study has been done on smart antennas, adaptive methods are not specially emphasized and developed. Adaptive beamforming is used for desired signal enhancement while suppressing interference at the output of an antenna array. In this work, different types of optimization techniques are used to form an adaptive system. Tuned LMS, Hybrid learning, RLS, harmony search and modified harmony search are the algorithms used to detect the angle of arrival and interference for multiple users. Comparative analysis of these techniques is also performed. A very high performance Xilinx FPGA device Virtex-5 XC5VSX50T is used. It generates multiple partial beams simultaneously using the offline generated optimized weights from a 5 element and 10 element array and their summation gives the full beam. Using FPGA virtex-5, optimized parameters from various adaptive algorithms are used to generate output signals and to track multiple users at the same instant and the results are observed in chipscope.

List of Figures

List of Tables

List of Abbreviations

Abbreviation	Description
AF	Array Factor
LMS	Least Mean Square
RLS	Recursive Least Square
HLA	Hybrid Learning Algorithm
HS	Harmony Search
MHS	Modified Harmony Search
ANFIS	Adaptive Neuro-Fuzzy Inference System
FPGA	Field-Programmable Gate Array
MSE	Mean Square Error
DOA	Direction Of Arrival
AOA	Angle Of Arrival
AOI	Angle Of Interference

Chapter 1

INTRODUCTION

1.1 Introduction

In recent years,there has been a requirement of more efficient use of radio spectrum. Smart antenna systems are capable of efficiently utilizing the radio spectrum and are an effective solution to the present wireless systems problems while achieving reliable and robust high speed, high data rate transmission. Smart antennas are arrays employing a set of radiating elements . The signals from these elements are combined to form a movable or switchable beam pattern that follows the desired user. The process of combining the signals and then focusing the radiation in a particular direction is often referred to as digital beamforming.The core of smart antenna is the adaptive beam- forming algorithms in antenna array. Adaptive Beamforming technique achieve maximum reception in a specified direction by estimating the signal arrival from a desired direction while signals of the same frequency from other directions are rejected. There are several Adaptive beamforming algorithms such as Tuned LMS, Hybrid learning algorithm, RLS, harmony search and modified harmony search varying in complexity based on different criteria for updating and computing the optimum weights. So it was not popular in the past due to the expensive cost of computation power. However, intensive signal processing is no longer an issue with the availability of low cost, extremely fast processors. It is more complicated when interfer-

ence from other mobile occurs. Though smart antenna techniques are new in the area of mobile communications, the technology itself was introduced in 1960s. The advent of powerful, low-cost, digital processing components and the development of software-based techniques has made smart antenna systems a practical reality for cellular communications systems.

1.2 Literature Review

For improving the capacity of the base station in wireless communication we need a antenna system which focus the radiated electromagnetic energy for improving the gain pattern[1]. One of the antenna model which suits above situation is Smart antenna. Main beam steering is not the only core technology of smart antenna but it can effectively reduce the multi-path interference and minimize the multi-path effect[2]. Different geometries of the adaptive antenna or smart antenna array are mentioned by El Zooghby, Ahmed in [3]. An array is analysed by considering the first element as the phase reference element. The objective function for the smart antenna synthesis is formed by Zaharis, Zaharias D and Skeberis, Christos and Xenos, Thomas D[4] .Successful design of smart antenna depends highly on the performance of DOA estimation algorithm as well as beamforming algorithm is shown and a systematic comparison of the performance of different DOA algorithm like Barlett, Capon, MUSIC has been extensively studied by analyzing the simulation result[5]. LMS incorporates an iterative procedure that makes successive corrections to the weight vector in the direction of the negative of the gradient vector which eventually leads to the minimum mean square error. The performance of the traditional LMS algorithm for different parameters is analysed[6]. The result obtained can achieve faster convergence and lower steady state error. Evaluation of the performance of LMS (Least Mean Square) beamforming algorithm in the form of normalized array factor (NAF) and mean square error(MSE) by varying the number of elements in the array and the placing between the sensor elements is done[7]. Re-

cursive Least Square (RLS) adaptive algorithm was used to compute the complex weights[8]. An artificial intelligence technique was used to optimize the parameters used in the design of rectangular microstrip patch antennas. This was achieved by using Adaptive Neuro-Fuzzy Inference System (ANFIS)[9]. This optimization method is simple, effective, and has low computer memory usage. The behaviour of an antenna array is nonlinear in nature, resulting in an extremely high complexity using this approach. A neural-network-based solution was provided to avoid complexity[10] by establishing a relation between the desired radiation patterns and feeding details such as voltage and spacing in the real antenna array converting the real array into a smart array.One of the recently developed evolutionary algorithm using the process of getting a perfect state of harmony by the musicians was conceptualized. Geem developed a new harmony search (HS) meta-heuristic algorithm that inspired from the musicians in 2001[11]. The harmony in music is analogous to the optimization solution vector, and the musicians improvisations are analogous to local and global search schemes in optimization techniques. The HS algorithm does not require initial values for the decision variables[12]. The HS algorithm has a novel stochastic derivative (for discrete variable) based on musicians experience, rather than gradient (for continuous variable) in differential calculus[13]. In optimization each decision variable initially chooses any value within the possible range, together making one solution vector. If all the values of decision variables make a good solution, that experience is stored in each variables memory, and the possibility of making a good solution is also increased next time. When a musician improvises one pitch, he (or she) has to follow some rules[14]. In HS, tuning of controlling parameters are very important to obtain the optimal solution[15]. For the tuning of parameter some modifications has been proposed. In 2007 a modification occurred in HS. In the modification in the creation of new harmony is done. Lucas M. Pavelski extended the use of HS to the multi-objectives[16]. A digital beam former was designed and im-

plemented and an architecture was developed for 4/8/12/16 element phased array radar[17]. This technique employed a very high performance FPGA to handle large number of mathematical computations. After suitable filtering, the channels are multiplied with RLS based optimized complex weights to form partial beams. The prototype architecture employed pipelining and parallelism techniques to generate multiple beams simultaneously from a 16 element array.

1.3 Objectives

The objective is to design the smart antenna array by employ ing five different algorithms like Tuned LMS ,Hybrid learning,RLS, harmony search and modified harmony search algorithm. Smart antennas are the antenna systems which can steer the main beam towards the desired direction called direction of arrival (DOA) and minimizing the intensity level at the interference angles. Now the objective is viewed as two main constraints steering the direction of arrival beam and other for minimizing the intensity at the interference level. The objective function is formulated as an optimization task of the angle of arrival (AOA) and other for angle of interference (AOI). Multiple number of angle of arrivals and angle of interferences are also detected at the same instant. In this project we are using FPGA Virtex-5 for this purpose.

This thesis work has following objectives as mentioned below:

- Modelling and formulation of Array Factor of linear antenna array
- Optimization of various system parameters using different algorithms. To compare the algorithms on the basis of error and array factor values
- Implementing in VHDL for real time application using the optimized parameters.
- Generating the output signals using fpga virtex-5 and observing the arrival and interference values with respect to time.

1.4 Thesis Organization

The organization of this thesis work is as follows:

- In chapter 2, a brief introduction of smart antenna systems with its types is presented. Further,linear antenna array with its mathematical array factor modelling is described. Also,implementation in FPGA device is discussed.

- In Chapter 3,different algorithms used in this thesis is described in detail with their mean square error and array factor curves.

- In Chapter 4,synthesis of linear antenna has been carried out and different algorithms are compared to obtain the best algorithm. Outputs from FPGA device with hardware setup is also discussed here.

- In chapter 5,thesis is concluded. Also,limitations along with future scope of this work is mentioned.

Chapter 2

SMART ANTENNA SYSTEMS

2.1 Antenna

An antenna is a device used for receiving or transmitting electromagnetic energy. Antennas couple electromagnetic energy from one medium (space) to another medium as wire, coaxial cable, or waveguide. Physical designs can vary greatly. Antenna produces complex electromagnetic fields both near to and far from antennas. Not all of the electromagnetic fields generated actually radiated into space. Some of the fields remain in the vicinity of antenna and are viewed as reactive near fields; much the same way as inductor or capacitor is a reactive storage element in lumped element circuits.

2.2 Smart Antenna

Smart antenna systems unite antenna technology and other technologies like digital signal processors. Smart antenna systems are providing higher rejection of interference,greater coverage area for each cell site, and substantial capacity improvements. That can overcome high speed mobile communication problems such as limited channel bandwidth while satisfying the demand for many users in a limited channel. In other words, such a system can automatically change the directionality of its radiation patterns in response to its signal environment.

2.3 Types of Smart Antenna Systems

Smart antenna systems are customarily categorized, however, as either switched beam or adaptive array systems. The following are distinctions between the two major categories of smart antennas regarding the choices in transmit strategy: Switched beam. A finite number of fixed, predefined patterns or combining strategies (sectors) Adaptive array. An infinite number of patterns (scenario-based) that are adjusted in real time.

2.3.1 Switched Beam Antennas

Switched beam antenna systems form multiple fixed beams with heightened sensitivity in particular directions. These antenna systems detect signal strength, choose from one of several predetermined, fixed beams, and switch from one beam to another as the mobile moves throughout the sector. Switched beam systems combine the outputs of multiple antennas in such a way as to form finely sectorized (directional) beams with more spatial selectivity than can be achieved with conventional, single element approaches.

2.3.2 Adaptive Array Antennas

The most advanced smart antenna approach to date is Adaptive antenna technology. Using a variety of new signal-processing algorithms, the adaptive system takes advantage of its ability to effectively locate and track various types of signals to dynamically minimize interference and maximize intended signal reception. Both systems attempt to increase gain according to the location of the user. However, only the adaptive system provides optimal gain while simultaneously identifying, tracking, and minimizing interfering signals.

2.4 Linear array

A linear arrangement shown in Fig.2.1 of N elements is considered which are uniformly distributed separated by a distance d.

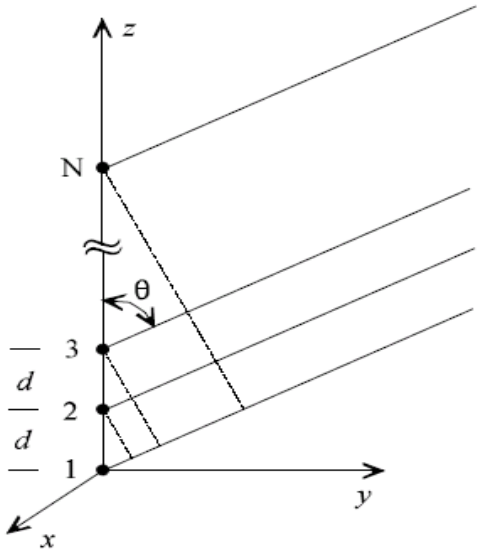

Figure 2.1: Uniform Linear Array

The formulation of the array factor is as follow. The current density value from the far field observation point is given by $J(\xi, \eta, \zeta)$. This current density is a parameter distributed in all three axes as below

$$J(\xi, \eta, \zeta) = J_x(\xi, \eta, \zeta) + J_y(\xi, \eta, \zeta) + J_z(\xi, \eta, \zeta) \tag{2.1}$$

where

$$J_x(\xi, \eta, \zeta) = J_x(x_i + \xi_i, y_i + \eta_i, z_i + \zeta_i)$$

The current density given in Eq.2.1 is radiated for the complex excitation. As the complex excitation changes the current density of the radiator will be varied[18]. Since all the elements are assumed to be identical and similarly oriented, the ratio of current density for the different complex excitations of the radiator follows,

$$\frac{J_x(x_i + \xi_i, y_i + \eta_i, z_i + \zeta_i)}{J_x(x_j + \xi_j, y_j + \eta_j, z_j + \zeta_j)} = \frac{I_i}{I_j} \tag{2.2}$$

With the aid of Eq.2.2 the far field equation with the linear arrangement over a finite volume is given as

$$A_\phi(\theta,\phi) = \sum_{i=0}^{N} \int_{V1} [-\sin\phi J_x(x_i+\xi_i, y_i+\eta_i, z_i+\zeta_i) + \cos\phi J_x(x_i+\xi_i, y_i+\eta_i, z_i+\zeta_i)] \exp^{jk\iota}, d\xi_i d\eta_i d\zeta_i \tag{2.3}$$

$$A_\theta(\theta,\phi) = \sum_{i=0}^{N} \int_{V1} [\cos\theta\cos\phi J_x(x_i+\xi_i, y_i+\eta_i, z_i+\zeta_i) + \cos\theta\sin\phi J_x(x_i+\xi_i, y_i+\eta_i, z_i+\zeta_i)] \exp^{jk\iota}, d\xi_i d\eta_i d\zeta_i \tag{2.4}$$

Where

$$\iota = \xi\sin\theta\cos\phi + \eta\sin\theta\sin\phi + \cos\theta$$

Considering the distance r_i for the i^th element from the far field observing point then the array factor for a linear field with field as above is

$$A_a(\theta,\phi) = \sum_{n=0}^{N} \frac{I_n}{I_0} \exp^{jkr_n(\cos\alpha\sin\theta\cos\phi + \cos\beta\sin\theta\sin\phi + \cos\gamma\cos\theta)} \tag{2.5}$$

2.5 Adaptive Antanna Array Modelling

2.5.1 Linear array modeling (Design Procedure)

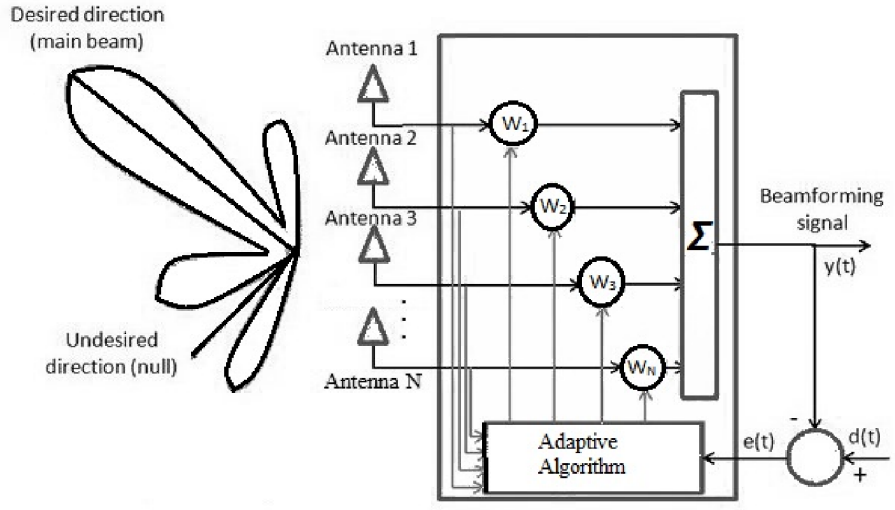

Figure 2.2: Adaptive Linear Array of isotropic elements

The Adaptive linear array itself clears the idea of adaptiveness. Weight estimation block in Fig.2.2 updates the weights depending on the output.
A uniformly spaced linear array of N = 5 isotropic antenna elements spaced a distance $d = \lambda/2$ in the X-axis can be depicted as:

$$a(\theta) = [1 \quad exp^{j\pi \sin\theta} \quad exp^{2j\pi \sin\theta} \quad exp^{3j\pi \sin\theta} \quad exp^{4j\pi \sin\theta}]$$

Parameter optimization:

1. LMS algorithm
2. Hybrid learning algorithm
3. RLS algorithm
4. Harmony search algorithm
5. Modified harmony search algorithm

2.5.2 Why FPGA

An FPGA based design is inherently parallel in nature. Different algorithm sequences operates concurrently which will be mapped to different hardware modules in a FPGA. The main reasons for choosing FPGA for generating smart antenna beams are as given below.

1)Parallel operation
2)Speed of execution
3)Flexibility
4)Low power design

2.5.3 Implementation in FPGA device

Here, Xilinx FPGA device Virtex-5 XC5VSX50T is used. The architecture generates multiple beams simultaneously from a 5 and 10 element array by employing techniques of pipelining and parallelism . The weights are calculated by using different algorithms and stored in the memory of the FPGA. For N number of beams, N different sets of sixteen weights are required. We consider the weights are fixed and calculated offline.Summation of all the

partial beams gives the full beam.The regenerated signal can be formulated as:

$$B(t) = \sum_{k=0}^{N} S_k(t) * W_k$$

where, N=Number of antenna elements,

S(t)= Received signal,

W_k=complex weight of kth element.

Chapter 3

BEAMFORMING ALGORITHMS

Adaptive Beamforming:
Adaptive Beamforming is a technique for controlling the direction of reception or transmission of a signal on an array. A standard tool for analyzing the performance of a beam-former is the response for a given N-by-1 weight vector W (k) as function of θ, known as the beam response. This angular response is computed for all possible angles.Various adaptive algorithms used in this work are explained below:

3.1 TUNED LEAST MEAN SQUARE ALGORITHM

The least mean square algorithm is based on the use of instantaneous values for cost function namely,

$$E(w) = \frac{e^2(n)}{2}$$

Where e(n) is the error signal measured at time n.

$$e(n) = d(n) - x^n w(n)$$

We may formulate lms algorithm as follows:

$$w(n+1) = w(n) + \eta x(n) e(n)$$

The feedback loop works like a low pass filter in LMS algorithm.The average time constant of this filtering section is inversely proportional to learning

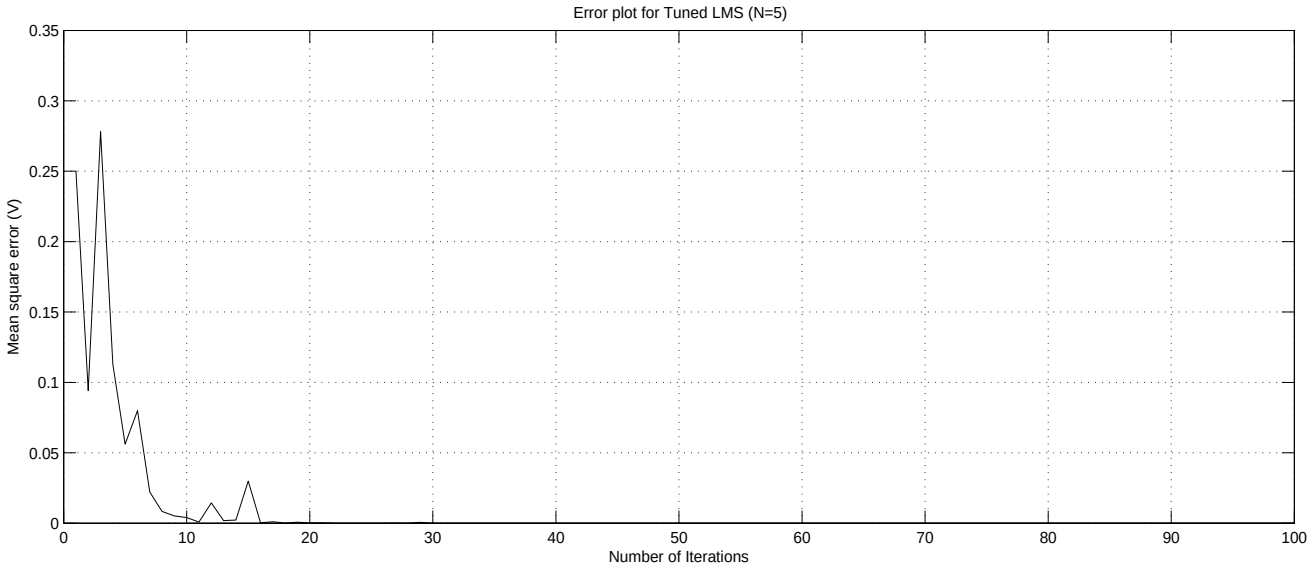

Figure 3.1: Error plot for Tuned LMS algorithm (N=5)

rate parameter. Hence the adaptive process will progress slowly by assigning small values to η[19].In tuned LMS,η is calculated using correlation matrix.

3.1.1 SIMULATION RESULTS OF TUNED LMS ALGORITHM

Simulation.I

Fig.3.1 shows the error plot for five elements using Tuned LMS algorithm.The mean square value of error after 100 iterations is 0.736e-11 V.

Simulation.II

Fig.3.2 shows the error plot for ten elements using Tuned LMS algorithm.The mean square value of error after 100 iterations is 0.438e-11 V.

Simulation.III

Fig.3.3 shows the array factor values in db for five elements at aoa=40^0,aoi=-40^0 using Tuned LMS algorithm. At aoa=40^0, we get array factor as -0.0019db.At aoi=-40^0,we get array factor as -50.936db.

Simulation.IV

Fig.3.4 shows the array factor values in db for ten elements at aoa=40^0,aoi=-40^0 using Tuned LMS algorithm. At aoa=40^0, we get array factor as 0 db.At aoi=-40^0,we get array factor as -60.477 db.

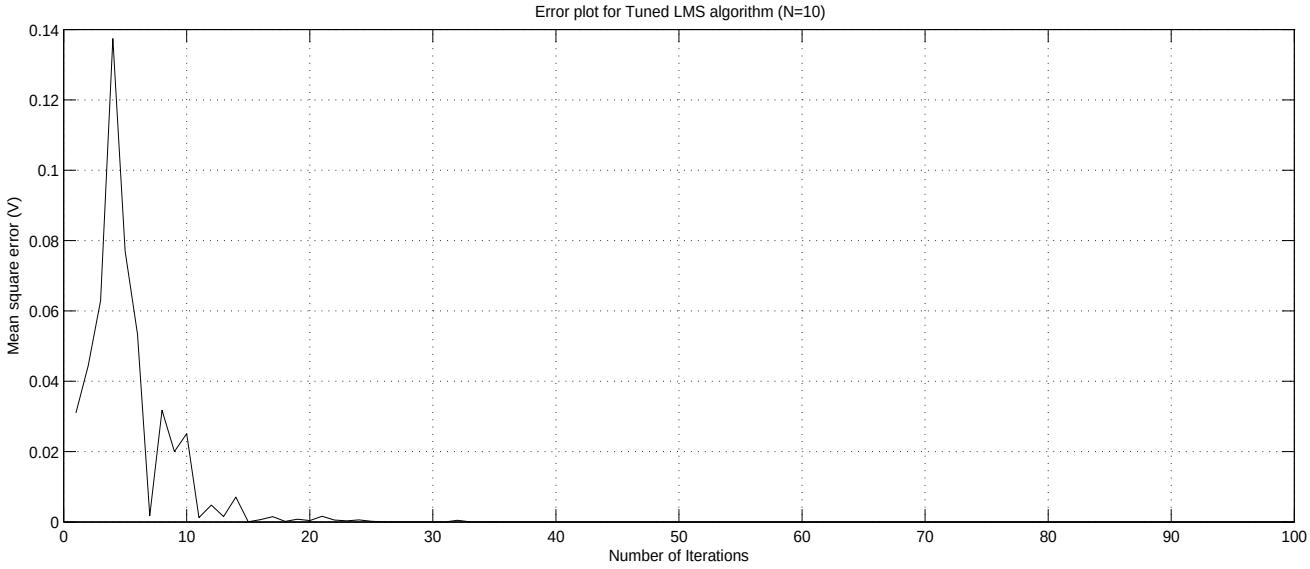

Figure 3.2: Error plot for Tuned LMS algorithm (N=10)

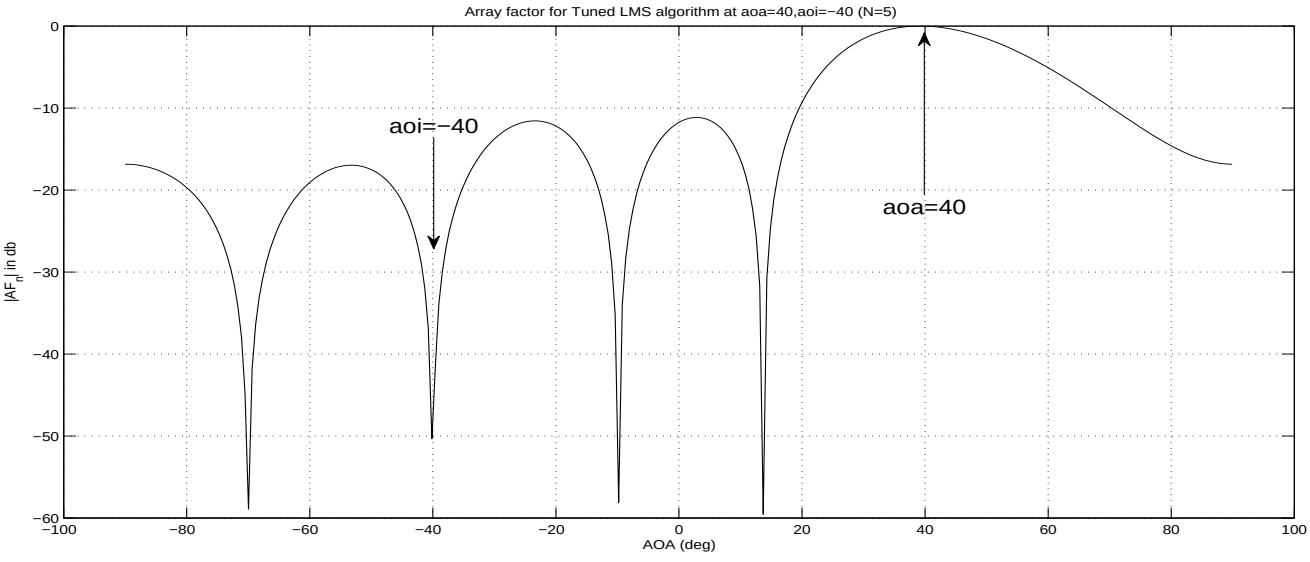

Figure 3.3: Array factor for Tuned LMS algorithm (N=5)

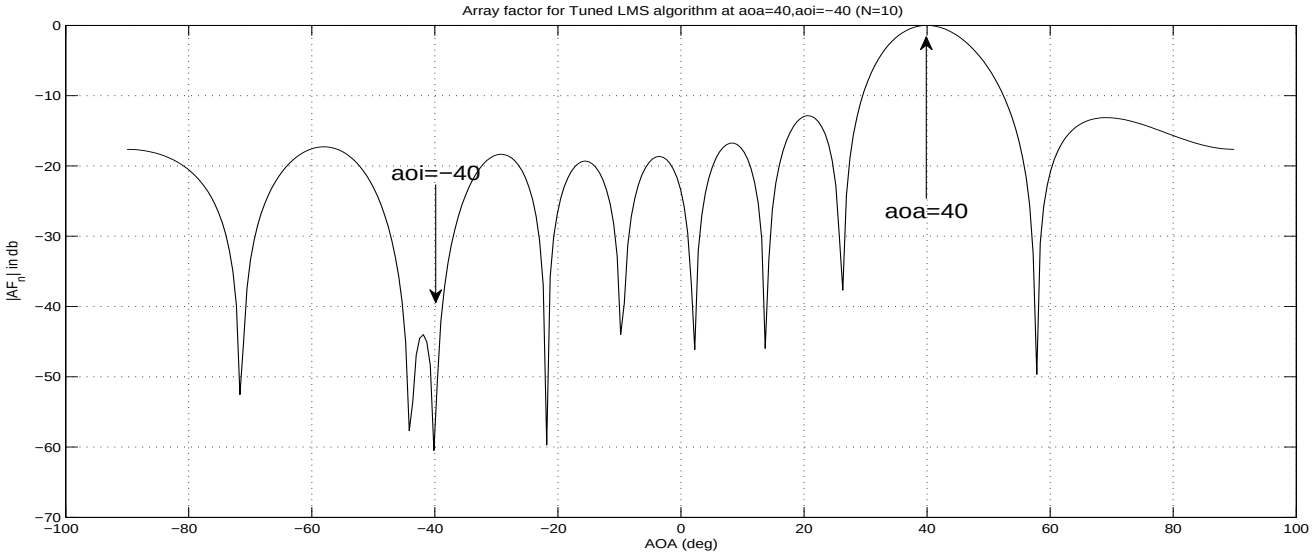

Figure 3.4: Array factor for Tuned LMS algorithm (N=10)

3.2 RECURSIVE LEAST SQUARE ALGORITHM

In recursive least-square (RLS) algorithm, the inverse correlation matrix is computed directly. The recursive least-squares (RLS) algorithm has a different approach to carry out the adaptation. The sum of the squared errors of different set of inputs is the subject of minimization instead of minimizing tne mean square error as in LMS algorithm.It requires reference signal and correlation matrix information[20]. In RLS algorithm the weights[21] are updated as follows:

$$w(k) = w(k-1) + g(k)[d(k) - x^H(k)w(k-1)]$$

Where g(k) is the gain vector and it expressed as

$$g(k) = R_{xx}^{-1}(k)x(k)$$

where

$$R_{xx}^k = \sum_{i=1}^{k} x(i)x^H(i)$$

.

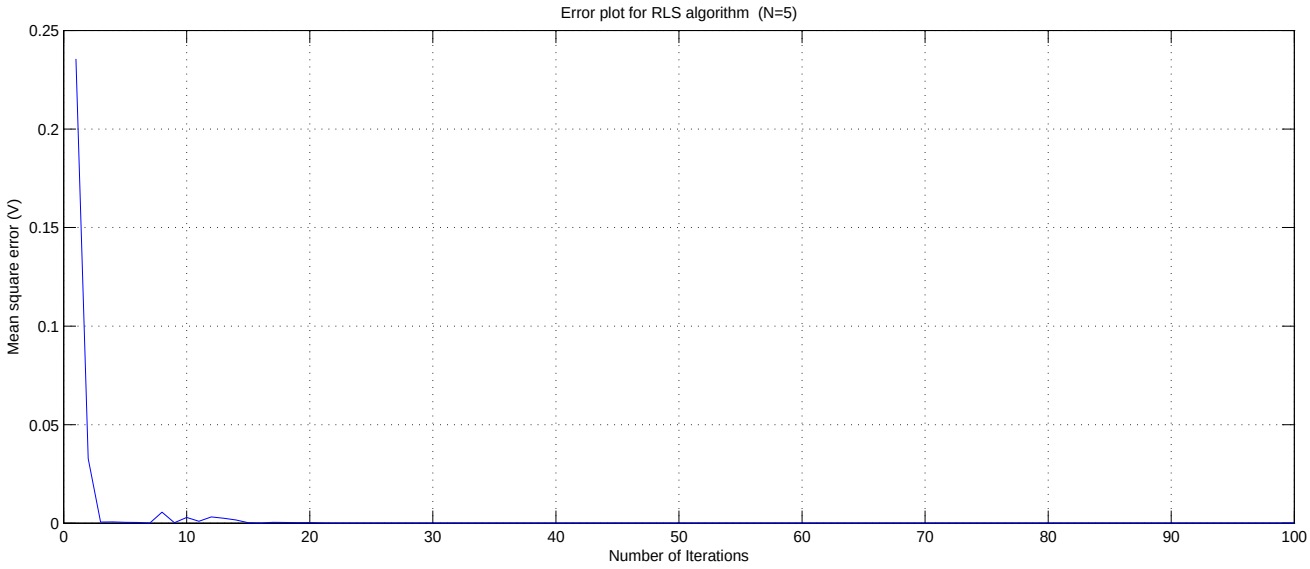

Figure 3.5: Error plot for RLS algorithm (N=5)

3.2.1 SIMULATION RESULTS OF RLS ALGORITHM

Simulation.I

Fig.3.5 shows the error plot for five elements using RLS algorithm.The mean square value of error after 100 iterations is 1.494e-11 V.

Simulation.II

Fig.3.6 shows the error plot for ten elements using RLS algorithm.The mean square value of error after 100 iterations is 2.858e-11 V.

Simulation.III

Fig.3.7 shows the array factor values in db for five elements at aoa=40^0,aoi=-40^0 using RLS algorithm. At aoa=40^0, we get array factor as -0.0641 db.At aoi=-40^0,we get array factor as -50.754 db.

Simulation.IV

Fig.3.8 shows the array factor values in db for ten elements at aoa=40^0,aoi=-40^0 using RLS algorithm. At aoa=40^0, we get array factor as 0 db.At aoi=-40^0,we get array factor as -57.758 db.

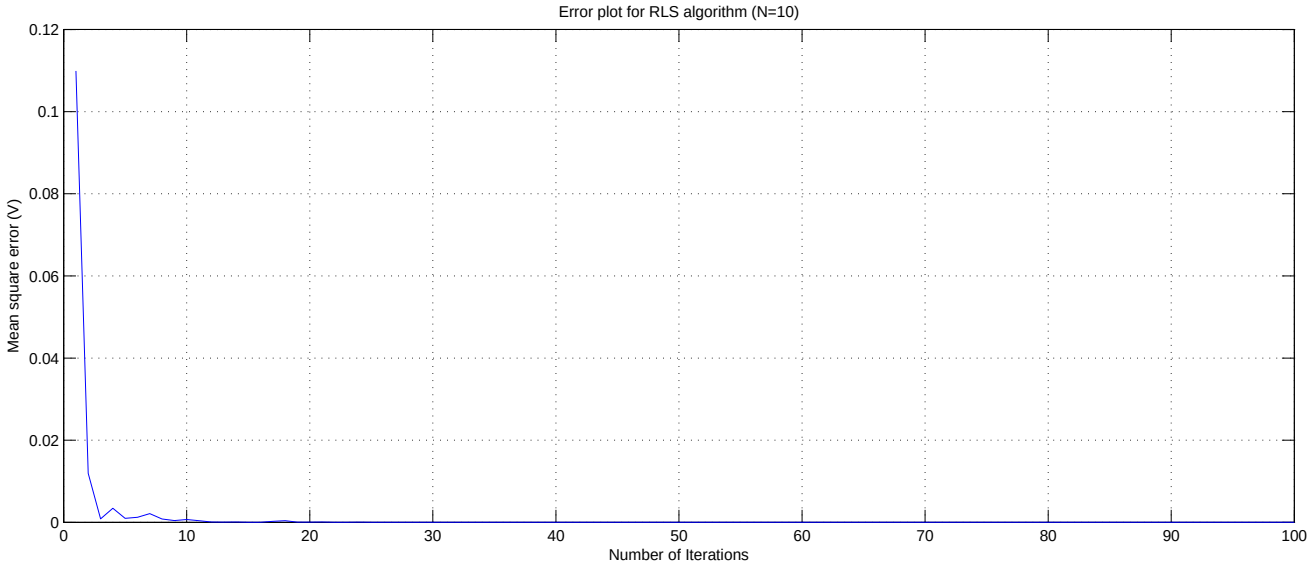

Figure 3.6: Error plot for RLS algorithm (N=10)

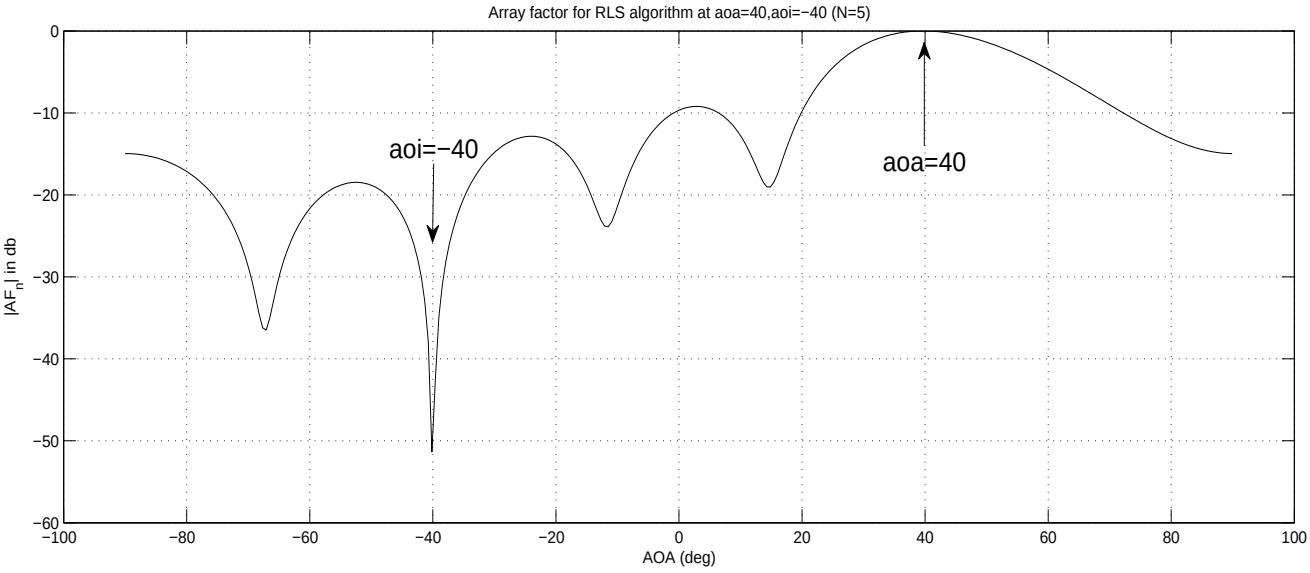

Figure 3.7: Array factor for RLS algorithm (N=5)

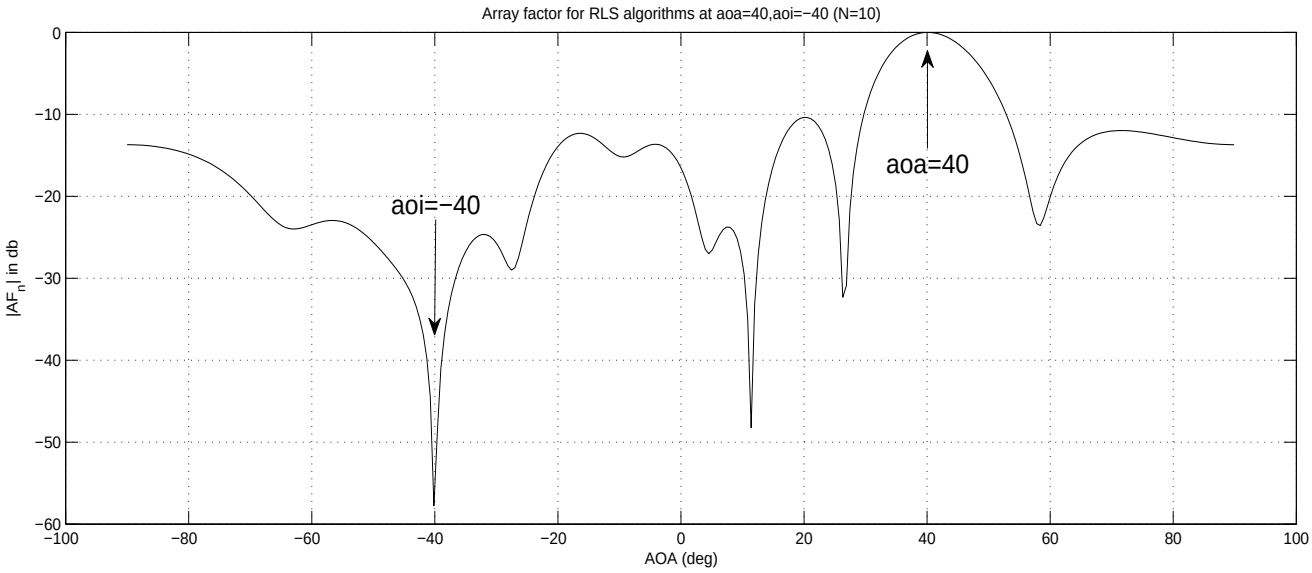

Figure 3.8: Array factor for RLS algorithm (N=10)

3.3 ANFIS (HYBRID LEARNING ALGORITHM)

The ANFIS provides a representation of the prior knowledge into a set of constraints (network topology) to reduce the optimization search space based on the fuzzy systems. An adaptive scheme is used for the fuzzy controlled parametric tuning based on the back-propagation mechanism popular in neural networks. The architecture of the ANFIS is explained[22]

Consider an ANFIS structure with n=2 inputs $[x_{1,k} \quad x_{2,k}]$ and m=2 mem-

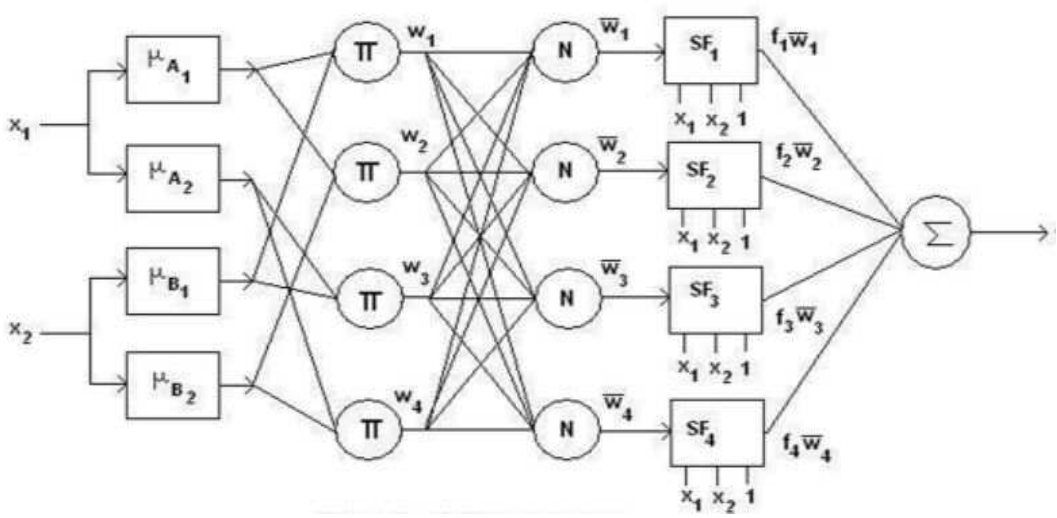

Figure 3.9: ANFIS Structure

bership functions (bell type) for each input. Let N=4 rule nodes be generated as depicted in the above figure. Hence there will be four sub-filter units in layer 4. Each sub-filter is composed of linear parameters$X_j = [p_j \quad q_j \quad r_j]$ with $1 \leq j \leq N$. The output of the nodes in each layer may be given as follows:

Layer 1:
The nodes of this layer contain the linguistic variables associated with the external input variables. The parameters of the membership functions (bell functions) representing the linguistic variables, $[a_{i,j} \quad b_{i,j} \quad c_{i,j}]$ are adaptive.The output of the nodes of this layer for an two inputs is given as:

$$\mu_{A_k}(x_{i,j}) = \frac{1}{1 + (\frac{x_{i,j} - c_{i,j}}{a_{i,j}})^{2b_{i,j}}}$$

where j=1,2;$1 \leq i \leq m$,and k=0,1,2,...,P-1,for P patterns per epoch of training data.

Layer 2:
This layer contains the fuzzy rule nodes which are implemented using the fuzzy conjunction operator or prod(.) . The node output in this layer is represented as:

$$w_j = \mu_{A_1}(x_{k,1}).\mu_{B_k}(x_{k,2})$$

Layer 3:
The nodes of this layer output the normalized output of the corresponding node in layer 2.

Layer 4:
This layer contains a sub-filter corresponding to each rule node. The sub-filter parameters $[p_j \quad q_j \quad r_j]$ are adaptive. The sub-filter operates as a linear combiner where the inputs are scaled by the parameters of the sub-filter and

finally added to give an output. The output of each sub-filter is expressed as

$$f_i\overline{w_i} = (\overline{w_i}x_{k,1}).p_i + (\overline{w_i}x_{k,2}).q_i + (\overline{w_i}).r_i$$

Layer 5:
This layer provides the network output which is expressed as the sum of the node outputs in layer 4.

$$f = \frac{\sum_i f_i w_i}{\sum_i w_i}$$

The ANFIS contains two layers (Layer 1 and Layer 4) of adaptive parameters. A hybrid learning rule is applied to train the ANFIS since it is composed of adaptive parameters belonging to two adaptive systems. The training is accomplished in two passes namely, the forward pass and the backward pass. In the forward pass the training data set (or epoch) is shown to the network while keeping the fuzzy parameters (otherwise called premise parameters) fixed. The error between the target value and the network output is calculated for each ensemble of the epoch and the parameters of the sub-filter (otherwise called consequent parameters) are updated recursively using the method of least squares.
In the backward pass, the parameters of the sub-filter are kept fixed while the parameters of the fuzzy membership functions are tuned using batch backpropagation method. This is done by calculating the network error for each pattern of the epoch and calculating the gradient of the output error square with respect to the parameters (using the chain rule) which are to be updated.

3.3.1 SIMULATION RESULTS OF HLA

Simulation.I

Fig.3.10 shows the error plot for five elements using hybrid learning algorithm . Membership fuction used is bell function. Number of premise and consequent parameters are 30 and 192 respectively. The mean square error value after 100 iterations is 7.951e-04 V.

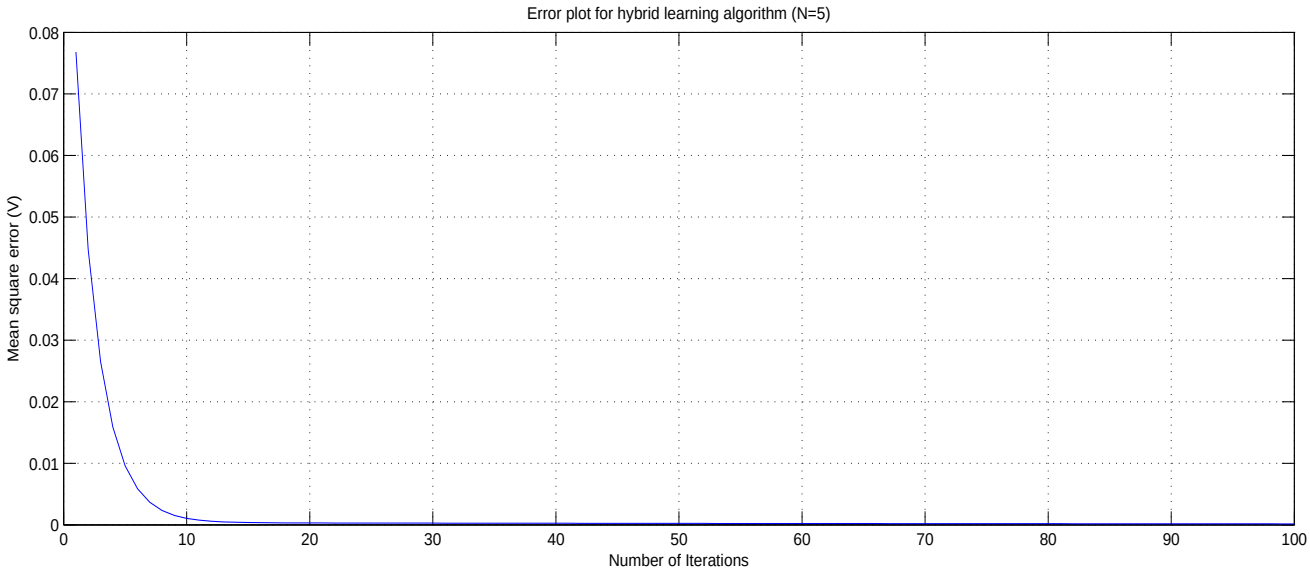

Figure 3.10: Error plot for Hybrid learning algorithm (N=5)

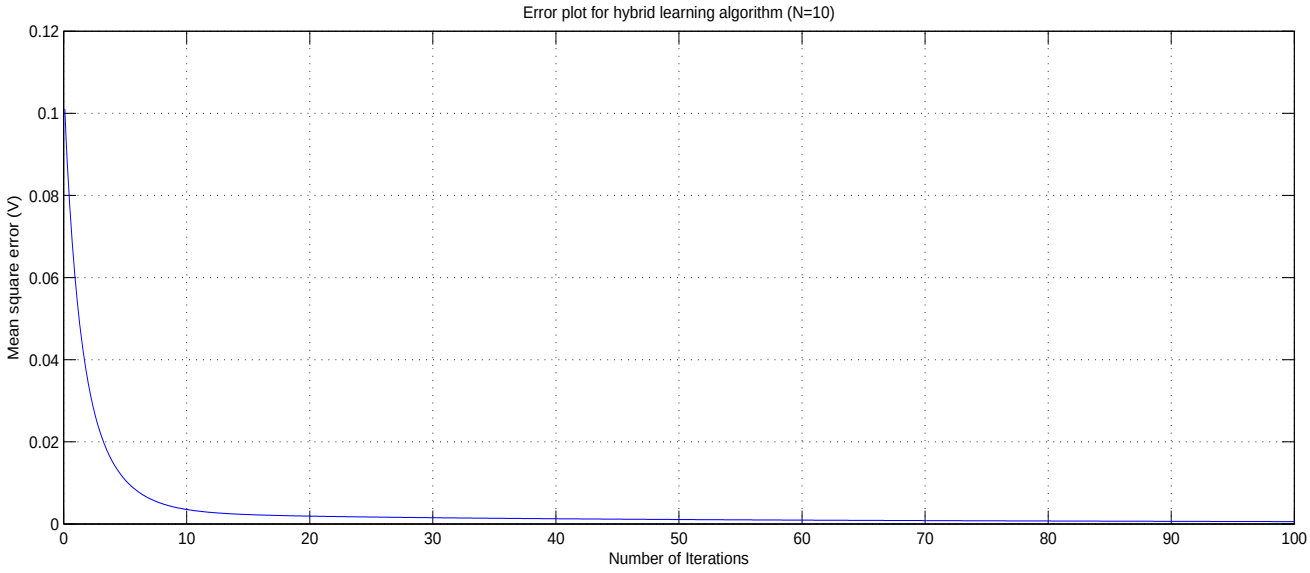

Figure 3.11: Error plot for Hybrid learning algorithm (N=10)

Simulation.II

Fig.3.11 shows the error plot for ten elements using hybrid learning algorithm. The mean square error value after 100 iterations is 4.568e-04 V.

Simulation.III

Fig.3.12 shows the array factor values in db for five elements at aoa=40^0,aoi=-40^0 using HLA. At aoa=40^0, we get array factor as -0.0051 db.At aoi=-40^0,we

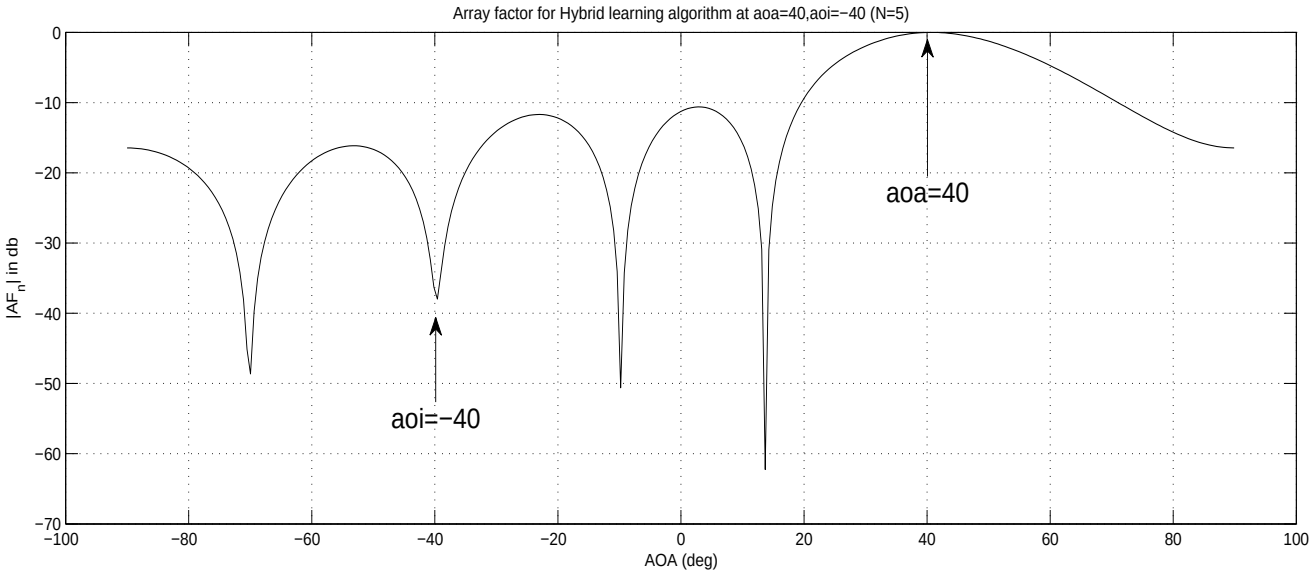

Figure 3.12: Array factor for Hybrid learning algorithm (N=5)

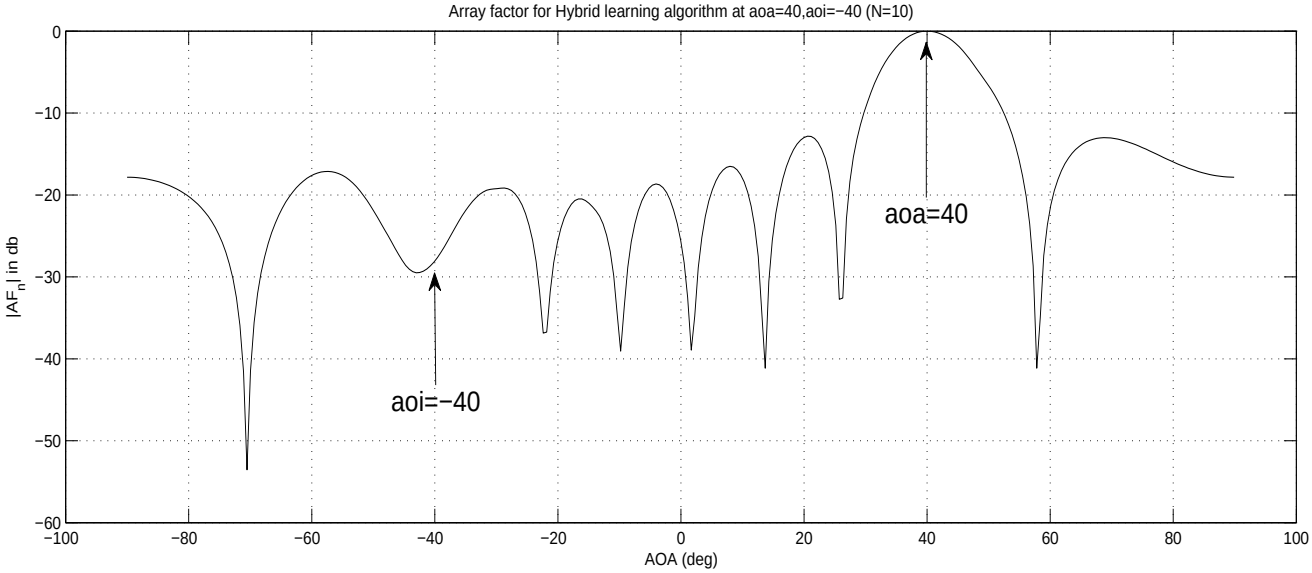

Figure 3.13: Array factor for Hybrid learning algorithm (N=10)

get array factor as -41.634 db.

Simulation.IV

Fig3.13 shows the array factor values in db for ten elements at aoa=40^0,aoi=-40^0 using HLA. At aoa=40^0, we get array factor as 0 db.At aoi=-40^0,we get array factor as -28.201 db.

3.4 HARMONY SEARCH

The HS is a met-heuristic algorithm which is inspired from the improvisation of a musician.Specifically the process by which the musicians rapidly refine their individual improvisation through variation resulting in an aesthetic harmony even if they have never played together. Here in our application each users treated as a musician and each instruments pitch and range corresponds to the bounds and constraints on the decision variable. The harmony between the musicians is taken as results when the audience aesthetic appreciation which is considered as cost function is in desired level. Every time musicians seek harmony over time through small variation to get the best cost[23].

When a decision variable chooses a value, it follows any one of three rules: (1) choosing any one value from the HS memory, (2) choosing an adjacent value of one value from the HS memory , and (3) choosing totally random value from the possible value range . This three rules are effectively directed in HS algorithm using two parameters, i.e., harmony memory considering rate (HMCR) and pitch adjusting rate (PAR).

3.4.1 SIMULATION RESULTS OF HS

Simulation.I

Fig.3.14 shows the error plot for five elements using HS algorithm.The mean square value of error after 100 iterations is 3.601e-05 V.

Simulation.II

Fig.3.15 shows the error plot for ten elements using HS algorithm.The mean square value of error after 100 iterations is 2.691e-05 V.

Simulation.III

Fig.3.16 shows the array factor values in db for five elements at aoa=40^0,aoi=-40^0 using HS algorithm. At aoa=40^0, we get array factor as -0.1487 db.At aoi=-40^0,we get array factor as -16.207 db.

Simulation.IV

Input Parameters $W = \begin{pmatrix} w_{1,1} & \cdots & w_{1,HMS} \\ \vdots & \ddots & \vdots \\ w_{N,1} & \cdots & w_{N,HMS} \end{pmatrix}$, HMS, PAR, HMCR
Output parameter w_o
Evaluate fitness $F = fitness(W)$, $F = \begin{bmatrix} f_1 & \dots & f_{HMS} \end{bmatrix}$
For iter < itermax
|
| if rand() < HMCR
| | W_n = rand(:,HMS)
| else
| | $W_n = W(:, i)$, i=any random integer
| end
| | if rand() < PAR
| | $W_n = W_n + BW$
| | BW = generally range of adjustment
| else
| | $W_n = W_n$+ random pitch
| end
| Evaluate fitness $F_n = fitness(W_n)$, $F = \begin{bmatrix} f_1 & \dots & f_H MS \end{bmatrix}$
| $W_o = \begin{cases} W_n & F_n < F \\ W & otherwise \end{cases}$
end

Table 3.1: HS PSEUDO Code

Parameter	Description
N	Number of Users
HMS	Harmony Memory Size ($50 or 100$)
CR	Considering Rate ($0.77 to 0.99$)
PAR	Pitch Adjusting Rate ($0.1 to 0.5$)
$ITERm$	Maximum Iterations
IT	Current iteration value
$RAND$	Random Variable Matrix
W	Weight Matrix
W_n	New Weight Matrix
F	Fitness values
F_n	New Fitness Value

Table 3.2: HS Parameter description

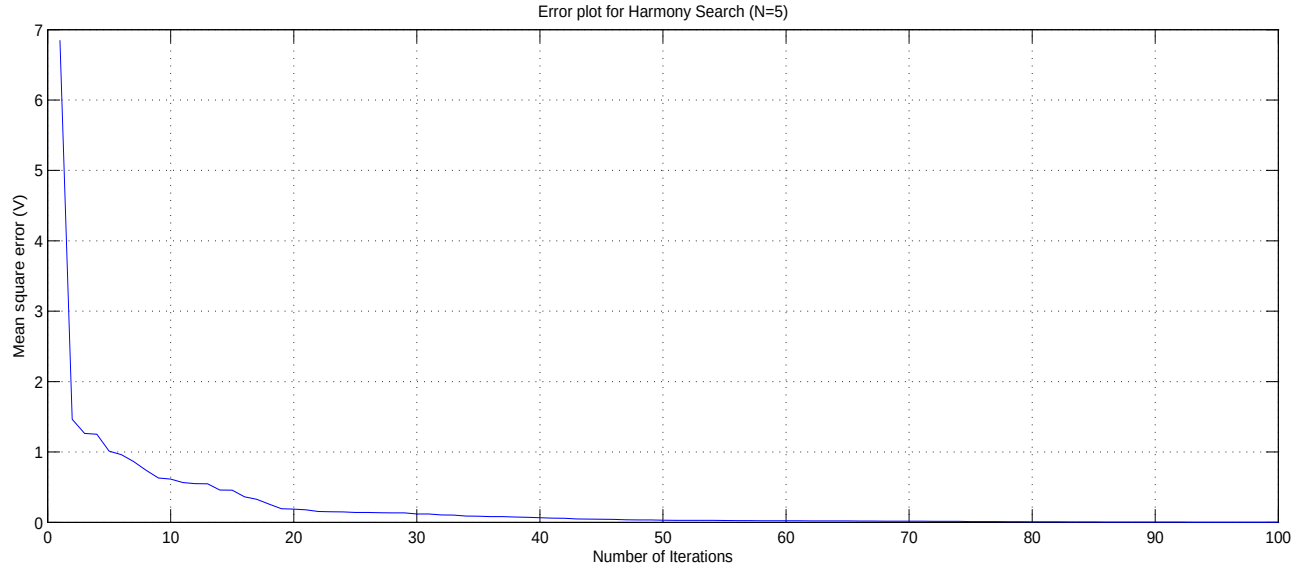

Figure 3.14: Error plot for Harmony search (N=5)

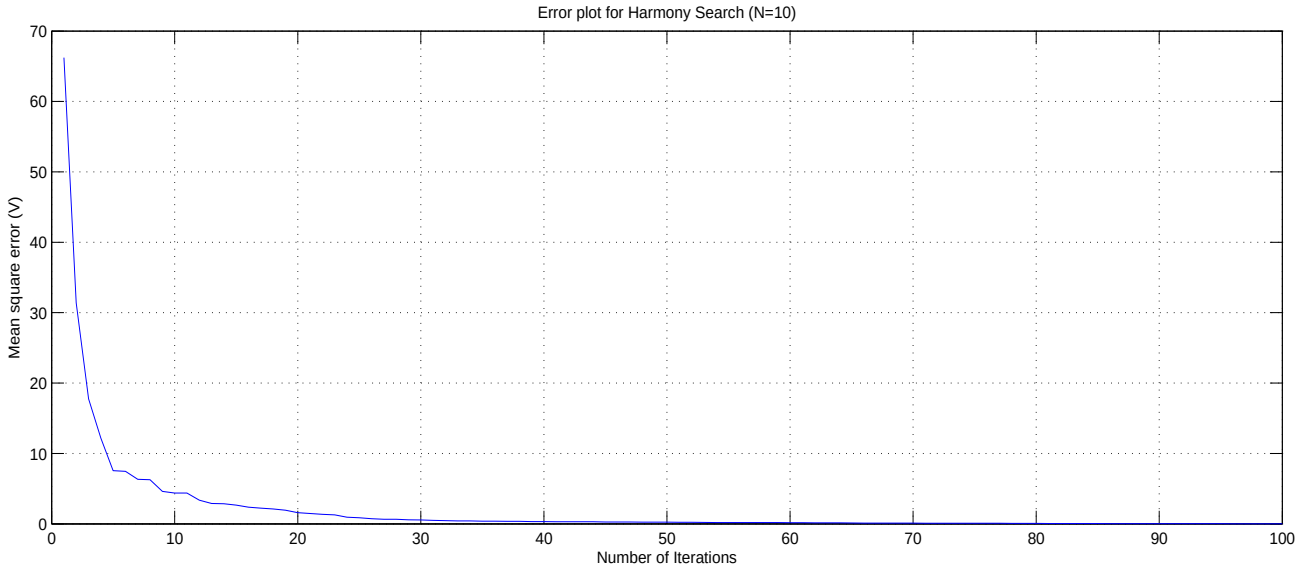

Figure 3.15: Error plot for Harmony search (N=10)

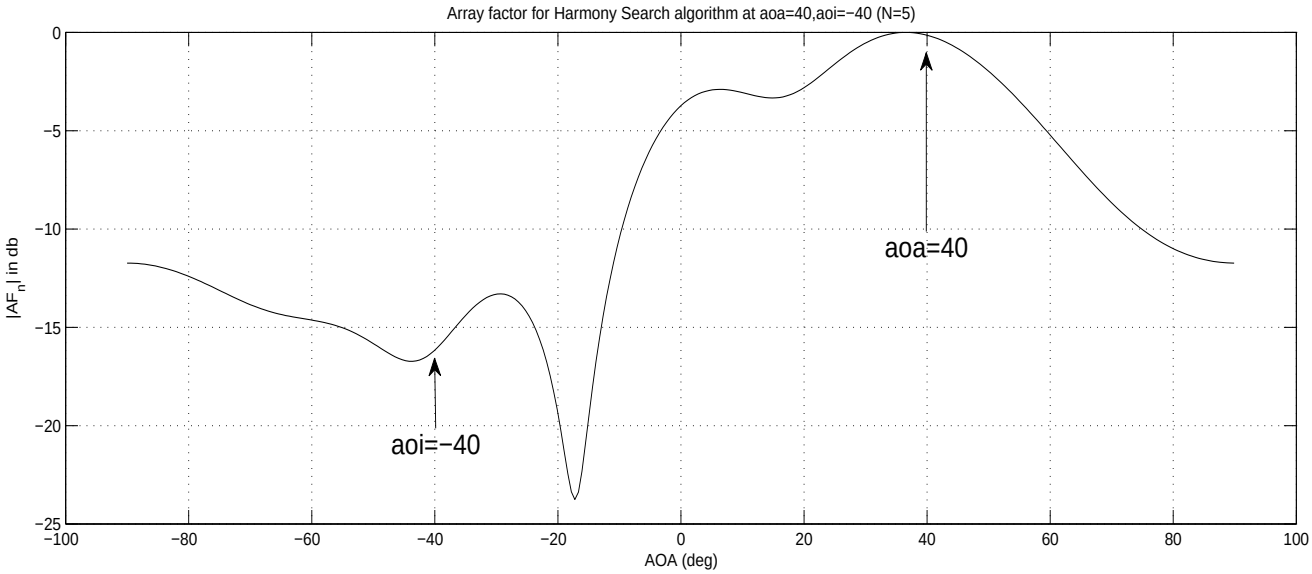

Figure 3.16: Array factor for harmony search (N=5)

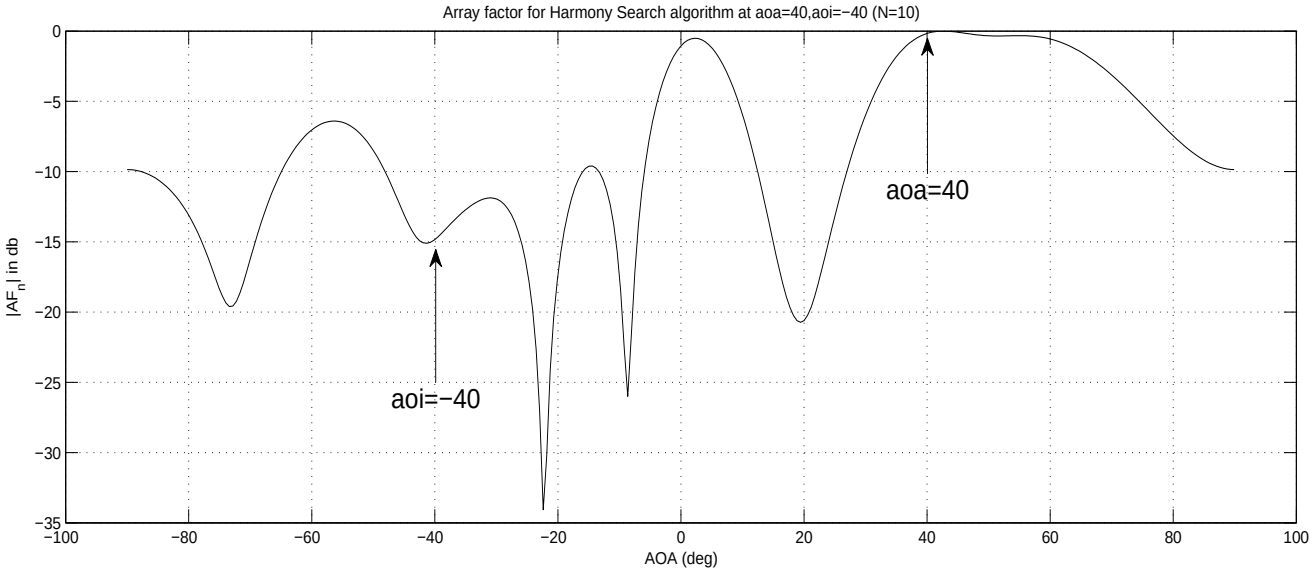

Figure 3.17: Array factor for harmony search (N=10)

Fig.3.17 shows the array factor values in db for ten elements at aoa=40^0,aoi=-40^0 using HLA. At aoa=40^0, we get array factor as -0.157 db.At aoi=-40^0,we get array factor as -14.684 db.

Input Parameters $W = \begin{pmatrix} w_{1,1} & \cdots & w_{1,HMS} \\ \vdots & \ddots & \vdots \\ w_{N,1} & \cdots & w_{N,HMS} \end{pmatrix}$, HMS, PAR=FCM(0-0.5),
HMCR=FCM(0.7-0.99)
Output parameter w_o
Evaluate fitness $F = fitness(W)$, $F = \begin{bmatrix} f_1 & \dots & f_{HMS} \end{bmatrix}$
For iter < itermax
|
| if rand() < HMCR
| | W_n = rand(:,HMS)
| else
| | $W_n = W(:, i)$, i=any random integer
| end
| | if rand() < PAR
| | $W_n = W_n + BW$
| | BW = generally range of adjustment
| else
| | $W_n = W_n$+ random pitch
| end
| Evaluate fitness $F_n = fitness(W_n)$, $F = \begin{bmatrix} f_1 & \dots & f_HMS \end{bmatrix}$
| $W_o = \begin{cases} W_n & F_n < F \\ W & otherwise \end{cases}$
| $HMCR(it) = \alpha - (\frac{(\beta + HMCR(it-1))}{1000} * it)$
| $PAR(it) = \gamma + \frac{(\frac{(\delta - PAR(it-1)}{1000} * it)}{HMS}$
where α,β,γ,δ are constants ranging from 0 to 1.
end

Table 3.3: MHS PSEUDO Code

3.5 MODIFIED HARMONY SEARCH

In Harmony Search,two parameters which dictates most impact on the performance of algorithm are HMCR and PAR.These parameters are introduced to allow the solution escape from local optima and reach/drive towards the global optimum prediction of the HS algorithm. Initially the values of HMCR and PAR are considered with in the range. This selection of parameters is purely random. In the Modified Harmony Search (MHS) the values of the HMCR and PAR are tuned/updated. This updation is mathematically represented in the following equations.

PROCESS	HS	MHS
size of populatioin (HMS)	User defined	user defined
Initial population (W)	Random	Random
Considering Rate ($HMCR$) initially	User defined	Value find from Fuzzy C-Mean clustering
Pitch Adjustment (PAR) initially	User defined	Value find from Fuzzy C-Mean clustering
Considering Rate ($HMCR$) after one iteration	Value fixed	Value changes depending on the its previous value
Pitch Adjustment (PAR) after one iteration	Value fixed	Value changes depending on the its previous value

Table 3.4: Comparison and Difference between HS and MHS

$$HMCR(it) = \alpha - (\frac{(\beta + HMCR(it-1))}{1000} * it) \tag{3.1}$$

$$PAR(it) = \gamma + \frac{(\frac{(\delta - PAR(it-1)}{1000} * it)}{HMS} \tag{3.2}$$

In the above equations represents the current iteration value. From the above equation it can be concluded that the updated value of the controlling parameters is dependent on the observed value in the previous iteration. Even this improvement suffers an escape from the global optima due to random initialization of the controlling parameters. This problem can be addressed with the support of fuzzy logic rule for initial setting. Fuzzy C-mean Clustering (FCM) is used for the initial selection of the control parameters. FCM will cluster the samples over the given range and gives a single best cluster value which is used as a initial set value for the HMCR and PAR. The pseudo code and changes/modification in HS heading to a new improved HS is described clearly.

3.5.1 SIMULATION RESULTS OF MHS

Simulation.I

Fig.3.18 shows the error plot for five elements using MHS algorithm.The mean square value of error after 100 iterations is 2.225e-07 V.

Simulation.II

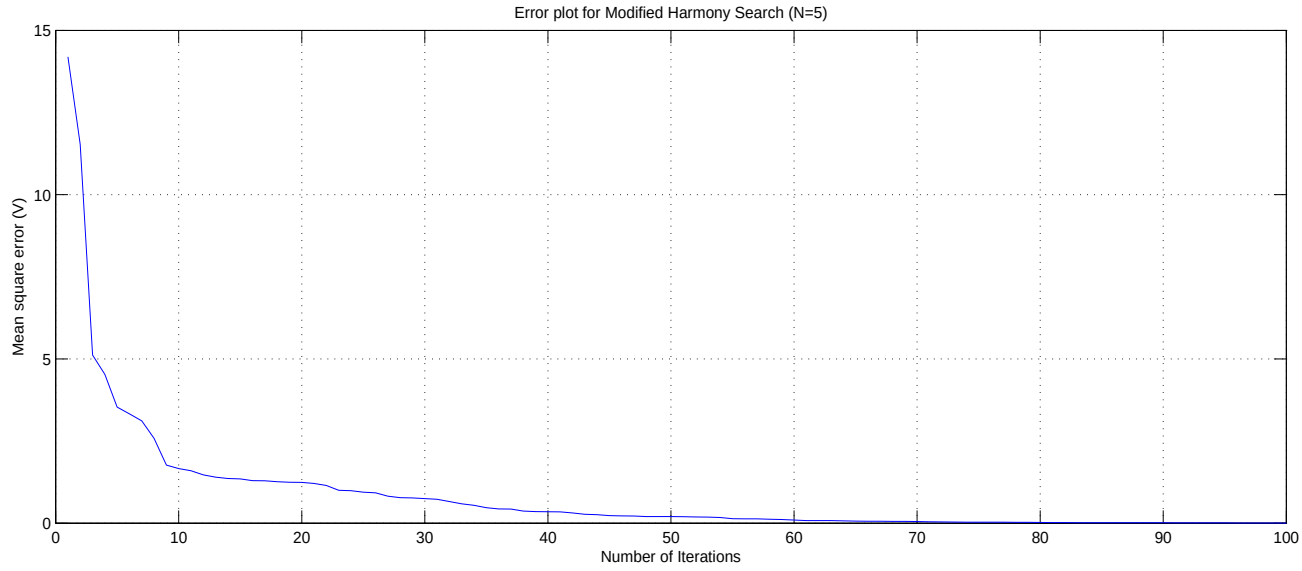

Figure 3.18: Error plot for Modified harmony search (N=5)

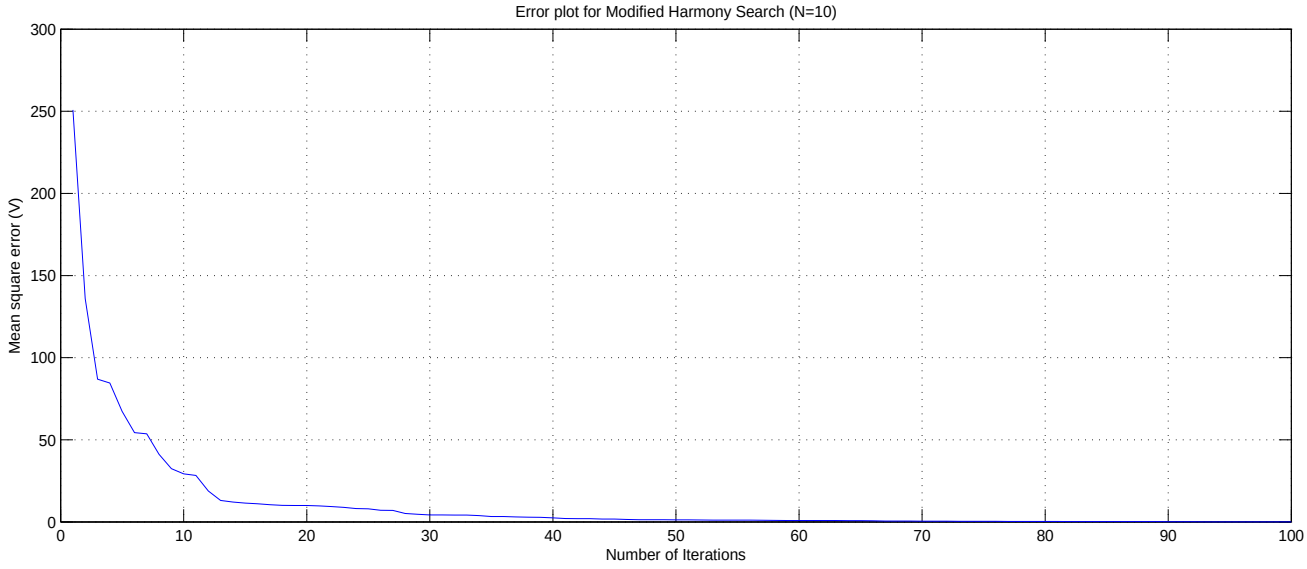

Figure 3.19: Error plot for Modified harmony search (N=10)

Fig.3.19 shows the error plot for ten elements using MHS algorithm.The mean square value of error after 100 iterations is 1.712e-07 V.

Simulation.III

Fig.3.20 shows the array factor values in db for five elements at aoa=40^0,aoi=-40^0 using MHS algorithm. At aoa=40^0, we get array factor as -0.0823 db.At aoi=-40^0,we get array factor as -11.067 db.

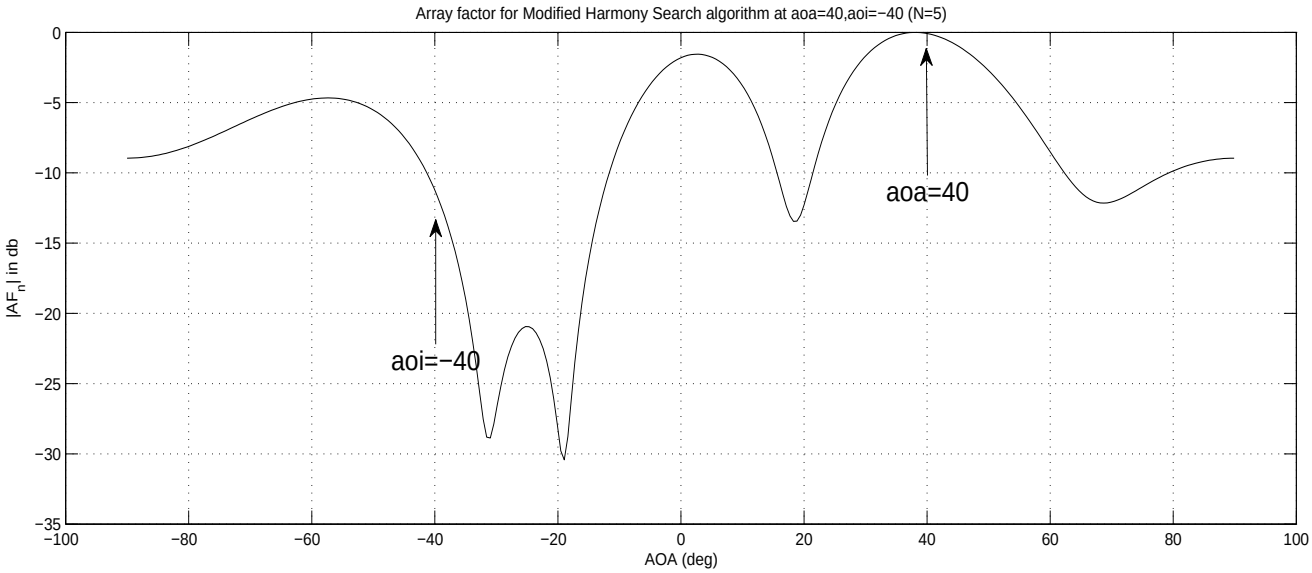

Figure 3.20: Array factor for Modified harmony search (N=5)

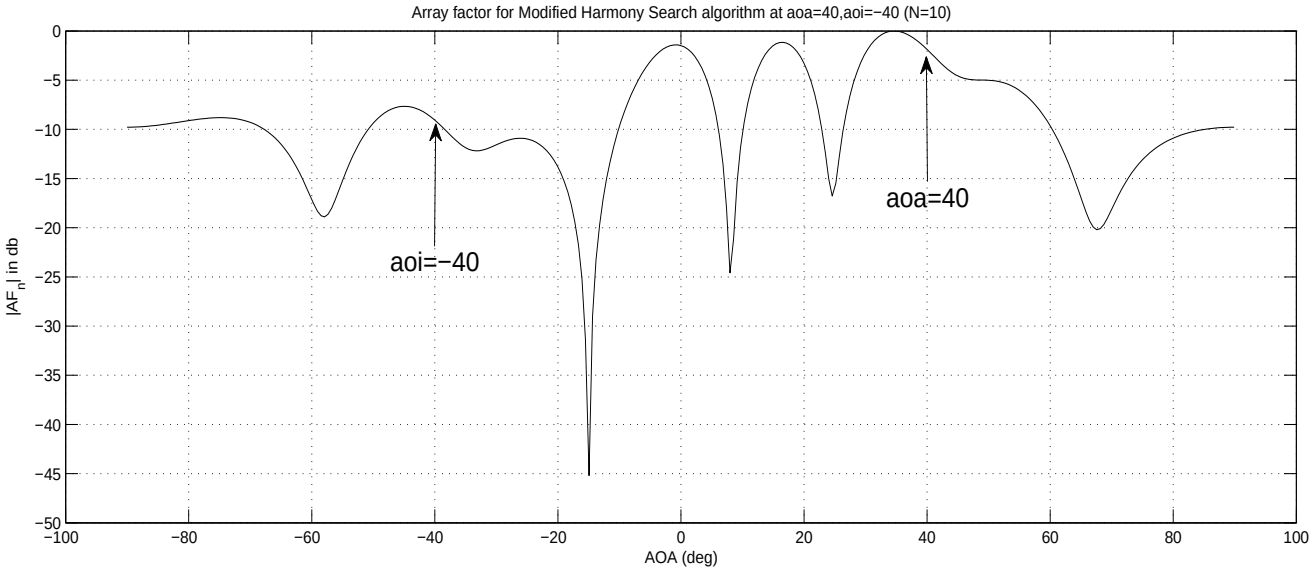

Figure 3.21: Array factor for Modified harmony search (N=10)

Simulation.IV

Fig.3.21 shows the array factor values in db for five elements at aoa=40^0,aoi=-40^0 using HLA. At aoa=40^0, we get array factor as -1.916 db.At aoi=-40^0,we get array factor as -8.976 db.

Chapter 4

SMART ANTENNA SYNTHESIS WITH RESULTS

The simulation results considering linear array geometry with aoa=40^0 and aoi=-40^0 using Tuned LMS,RLS,HLA,HS and MHS are as follows:

Algorithm	N	Iterations	Error (V)		Array factor (db)	
			Best	Worst	AOA	AOI
Tuned LMS	5	100	0.736e-11	0.278	-0.0019	-50.936
RLS	5	100	1.494e-11	0.234	-0.0641	-50.754
Hybrid Learning Algorithm	5	100	7.951e-04	0.0768	-0.0051	-41.634
HS	5	100	3.601e-05	7.25	-0.1487	-16.207
MHS	5	100	2.225e-07	14.711	-0.0823	-11.067

Table 4.1: Comparison between different algorithms (N=5)

Algorithm	N	Iterations	Error (V)		Array factor (db)	
			Best	Worst	AOA	AOI
Tuned LMS	10	100	0.438e-11	0.137	0	-60.477
RLS	10	100	2.858e-11	0.109	0	-57.758
Hybrid Learning Algorithm	10	100	4.5e-04	0.103	0	-28.201
HS	10	100	2.691e-05	66.24	-0.157	-14.684
MHS	10	100	1.712e-07	250.67	-1.916	-8.976

Table 4.2: Comparison between different algorithms (N=10)

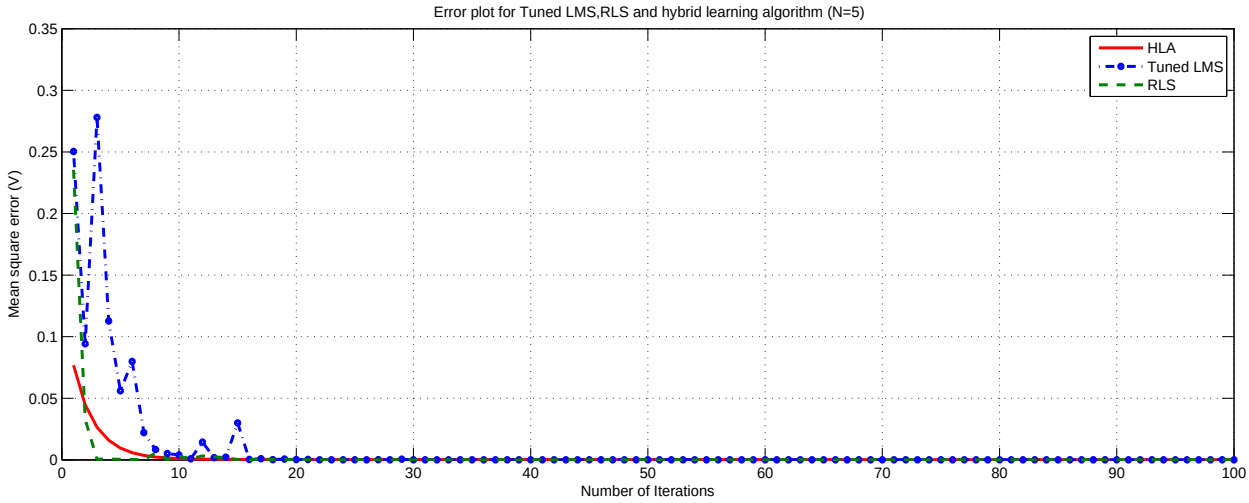

Figure 4.1: Error plot for Tuned LMS,RLS and HLA (N=5)

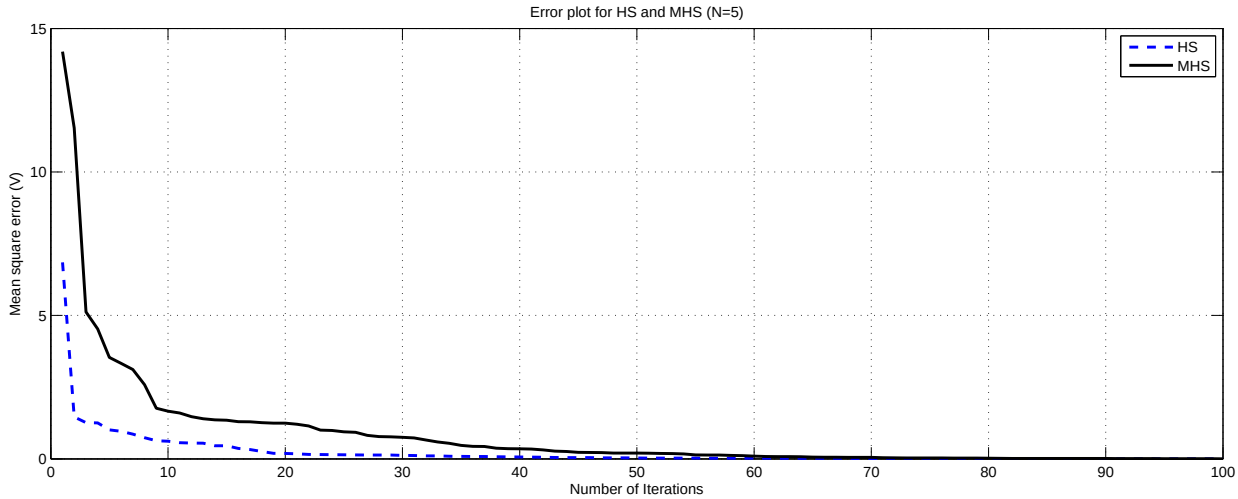

Figure 4.2: Error plot for HS and MHS (N=5)

4.1 Error plots

Simulation.I and II

Fig.4.1 and Fig.4.2shows the error plot for five elements using Tuned LMS,RLS,HLA, HS and MHS for 100 iterations.Tuned LMS algorithm gives the best mean square error value after 100 iterations as 0.736e-11 V .

Simulation.III and IV

Fig.4.3 and Fig.4.4 shows the error plot for ten elements using Tuned LMS,RLS,HLA,

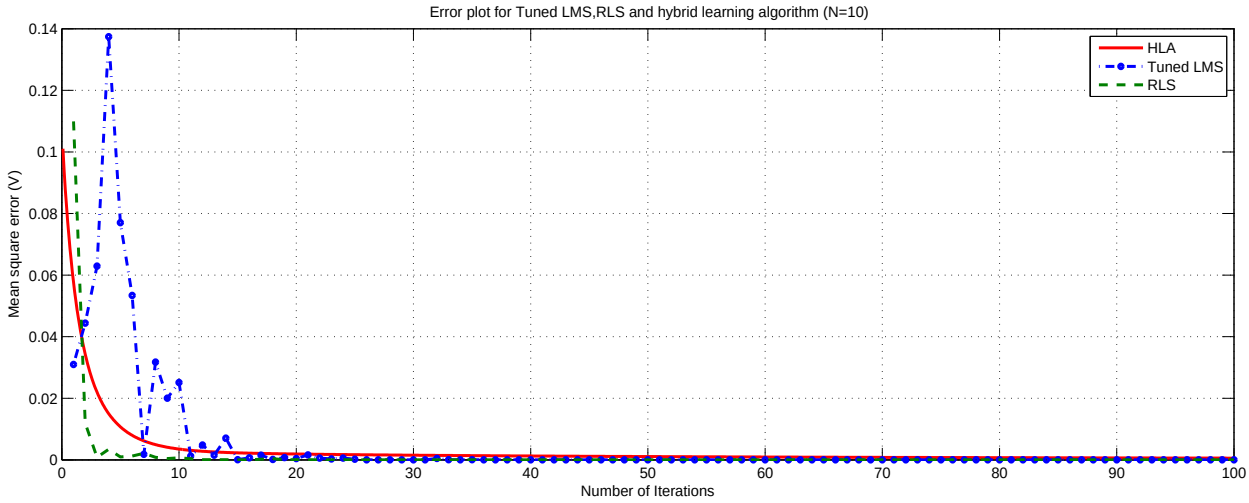

Figure 4.3: Error plot for Tuned LMS,RLS and HLA (N=10)

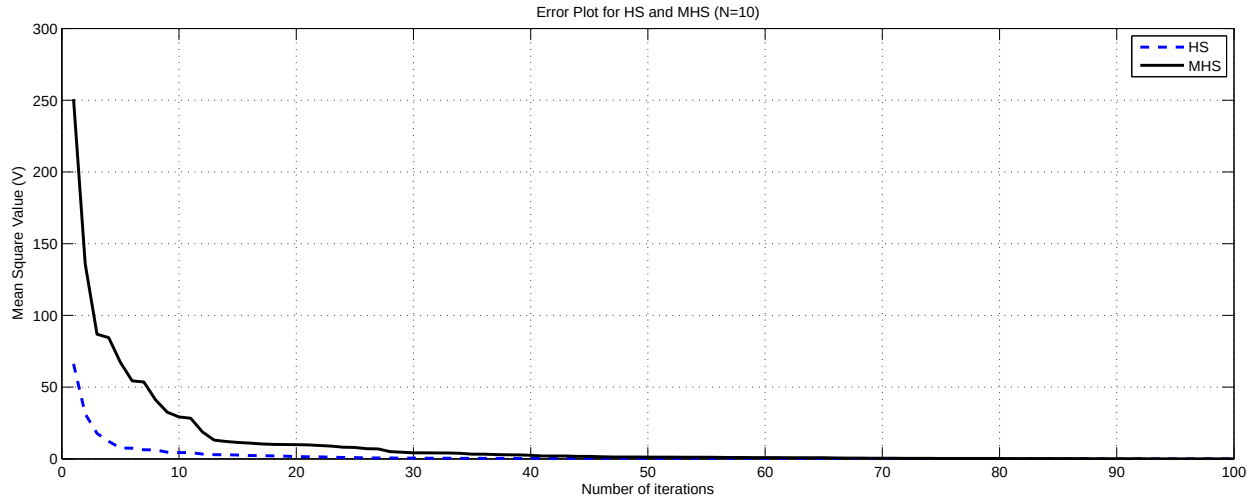

Figure 4.4: Error plot for HS and MHS (N=10)

HS and MHS for 100 iterations.Here also, Tuned LMS algorithm gives the best mean square error value after 100 iterations as 0.438e-11 V .

4.2 Regenerated signals

Simulation.I

Fig.4.5shows the regenerated signal for different algorithms when number of array elements is 5.

Simulation.II

Fig.4.6 shows the regenerated signal for different algorithms when number of

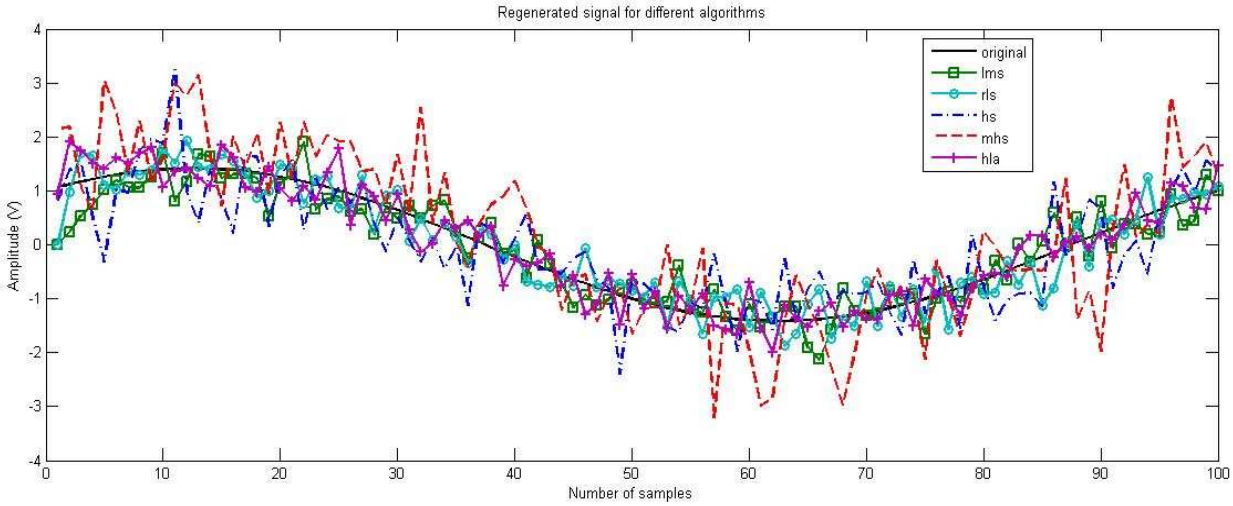

Figure 4.5: Regenerated signals for N=5

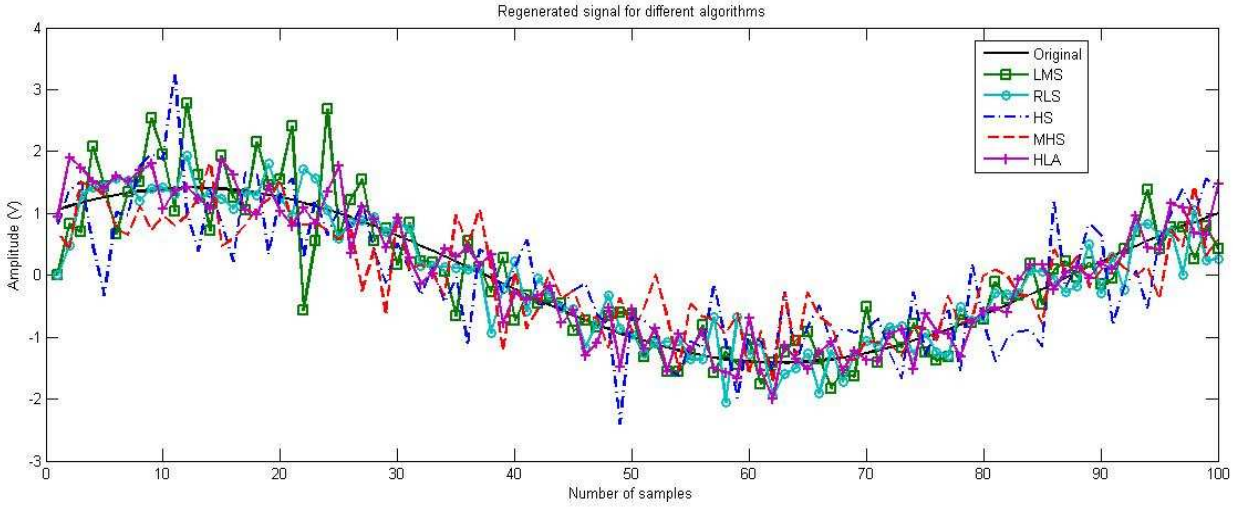

Figure 4.6: Regenerated signals for N=10

array elements is 10.

4.3 Array factor plots for single angle of arrival

Simulation.I

Fig.4.7 shows the array factor values in db for five elements at aoa=40^0,aoi=-40^0 using Tuned LMS,RLS,HLA,HS and MHS. For Tuned LMS algorithm,at aoa=40^0, we get array factor as -0.0019b.At aoi=-40^0,we get array factor as -50.936db.Comparing with other algorithms,Tuned LMS gives better array

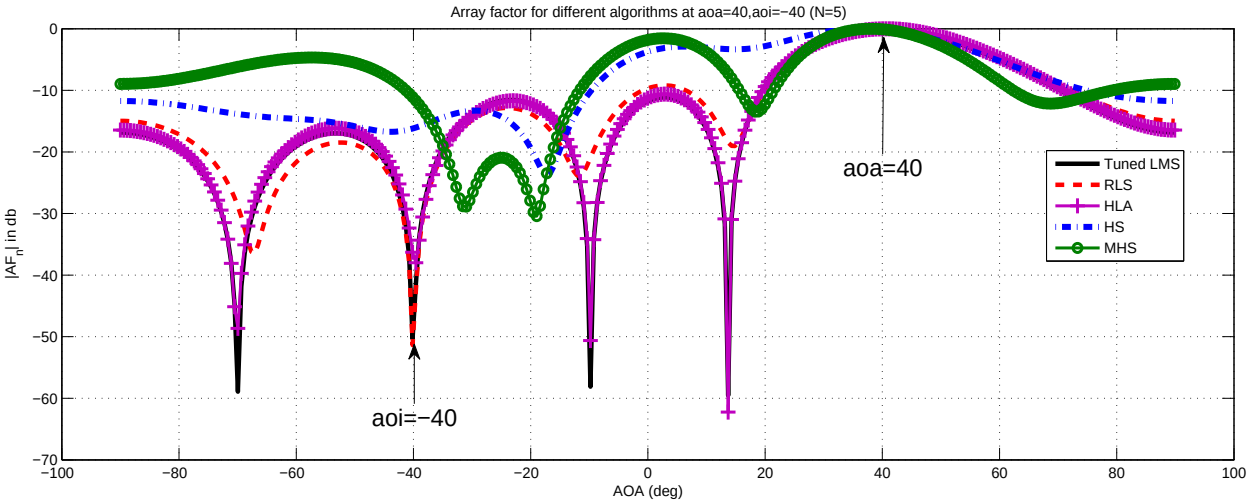

Figure 4.7: Array factor for different algorithms (N=5)

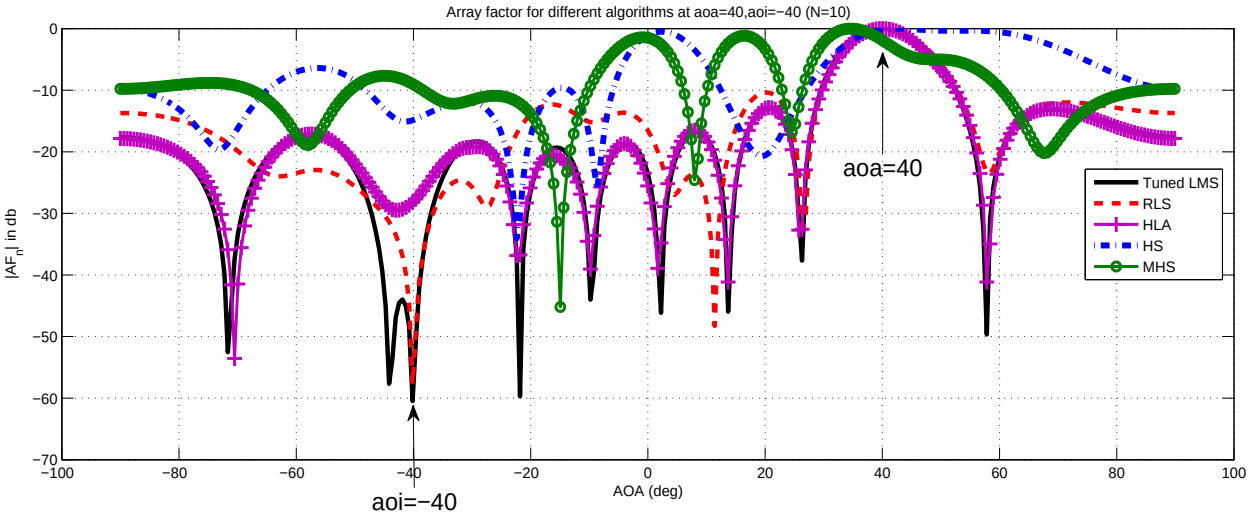

Figure 4.8: Array factor for different algorithms (N=10)

factor values.

Simulation.II

Fig.4.8 shows the array factor values in db for ten elements at aoa=40^0,aoi=-40^0 using Tuned LMS,RLS,HLA,HS and MHS. For Tuned LMS algorithm,at aoa=40^0, we get array factor as 0db.At aoi=-40^0,we get array factor as -60.477db.Comparing with other algorithms,Tuned LMS gives better array factor values.

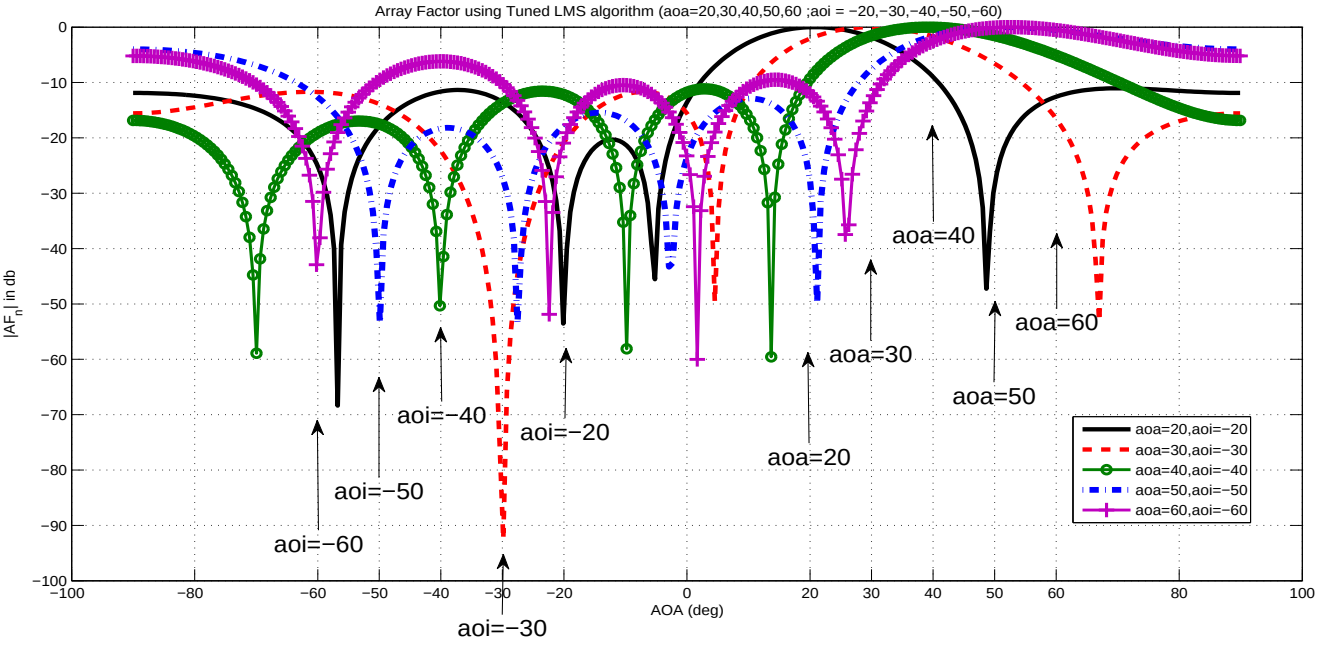

Figure 4.9: Array factor for five angle of arrivals using Tuned LMS (N=5)

4.4 Array Factor plots for multiple angle of arrivals and interferences

4.4.1 Using Tuned LMS algorithm

Simulation.I

Fig.4.9 shows the array factor values in db for five elements at five angle of arrival and interferences using Tuned LMS. At aoa=20^0,30^0,40^0,50^0,60^0, we get array factor as -0.0115db, -0.0007db, -0.0019db, -0.0093db, -0.4515db respectively.At aoi=-20^0,-30^0,-40^0,-50^0,-60^0,we get array factor as -53.556db, -79.994db, -50.936db, -40.459db, -42.905db respectively.

Simulation.II

Fig.4.10 shows the array factor values in db for ten elements at five angle of arrival and interferences using Tuned LMS. At aoa=20^0,30^0,40^0,50^0,60^0, we get array factor as 0db, -0.0032db, 0db, 0db, 0db, respectively.At aoi=-20^0,-30^0,-40^0,-50^0,-60^0,we get array factor as -65.187db, -40.208db,-60.477db, -45.822db, -53.374 respectively.

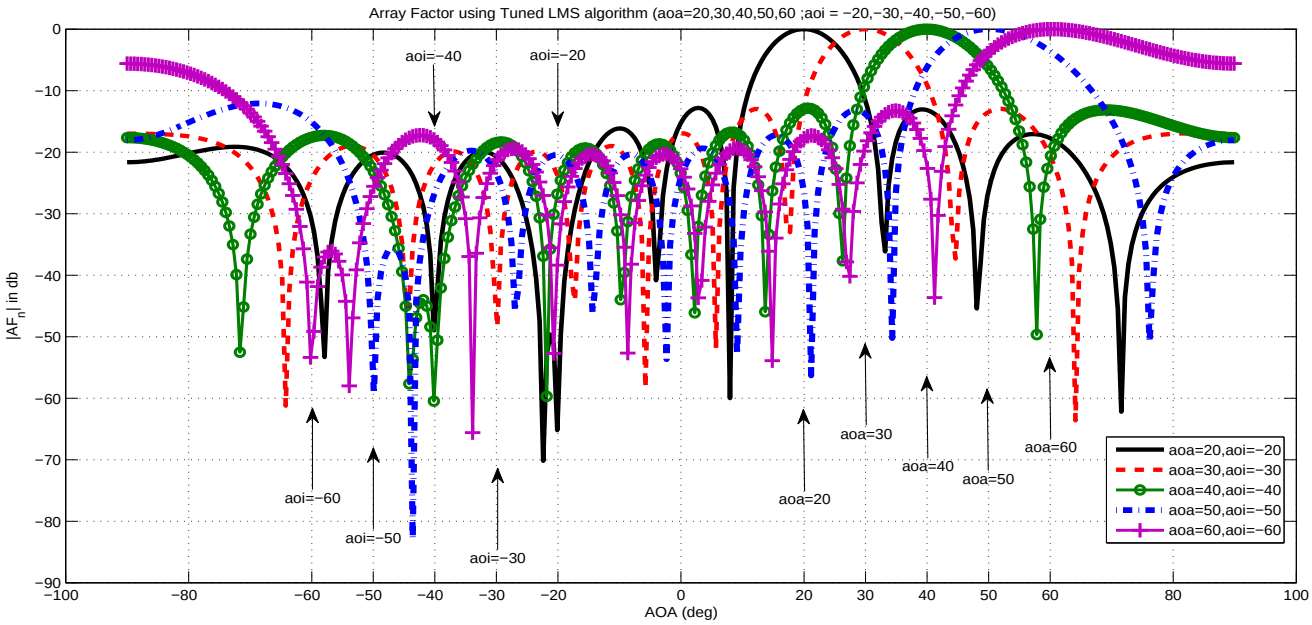

Figure 4.10: Array factor for five angle of arrivals using Tuned LMS (N=10)

4.4.2 Using RLS algorithm

Simulation.I

Fig.4.11 shows the array factor values in db for five elements at five angle of arrival and interferences using RLS. At aoa=20^0,30^0,40^0,50^0,60^0, we get array factor as -0.1117db, -0.0149db, -0.0641db, -0.0613db, -0.5471db respectively.At aoi=-20^0,-30^0,-40^0,-50^0,-60^0,we get array factor as -50.562db, -61.147db, -50.754db, -39.309db, -40.670db respectively.

Simulation.II

Fig.4.12 shows the array factor values in db for ten elements at five angle of arrival and interferences using RLS. At aoa=20^0,30^0,40^0,50^0,60^0, we get array factor as -0.0371db, -0.048db, 0db, 0db, -0.0043db respectively.At aoi=-20^0,-30^0,-40^0,-50^0,-60^0,we get array factor as -60.304db, -36.743db, -57.758db, -42.101db, -52.753db respectively.

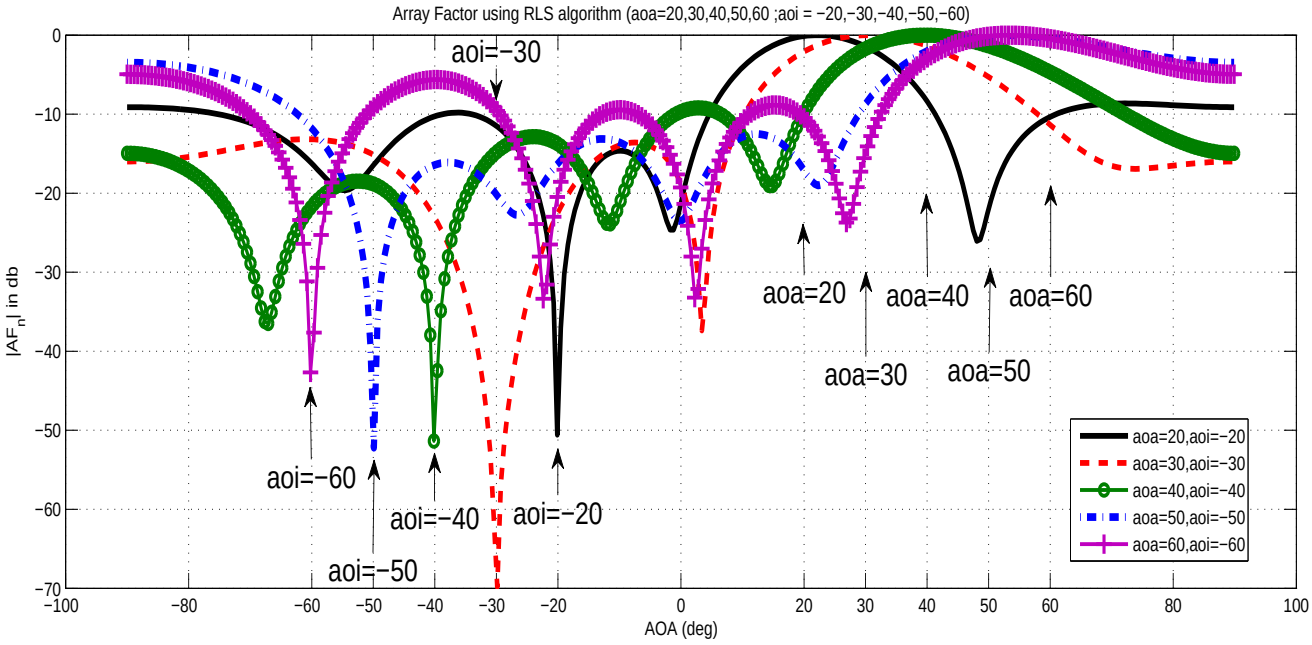

Figure 4.11: Array factor for five angle of arrivals using RLS (N=5)

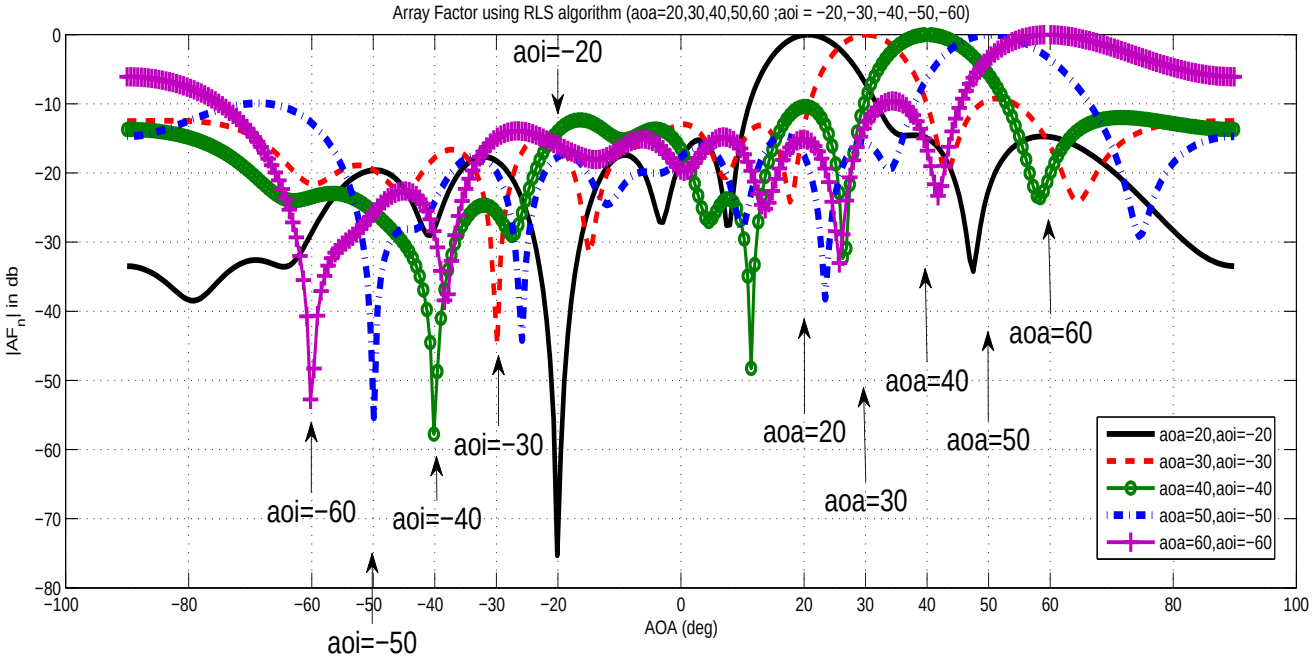

Figure 4.12: Array factor for five angle of arrivals using RLS (N=10)

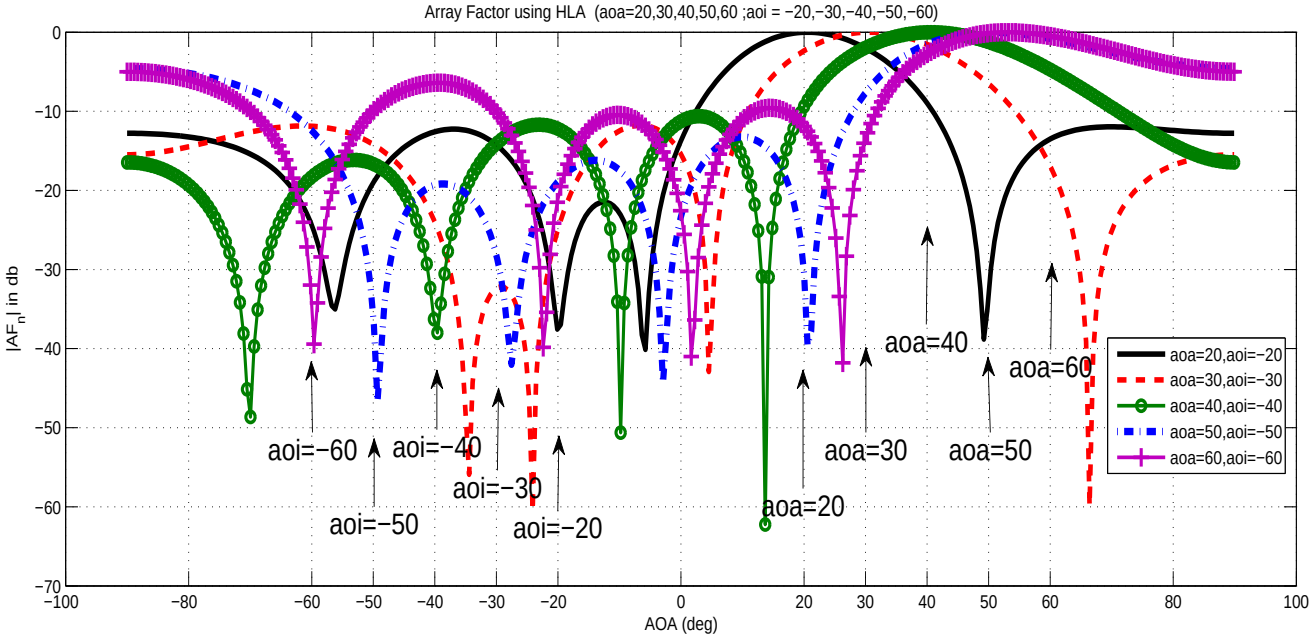

Figure 4.13: Array factor for five angle of arrivals using HLA (N=5)

4.4.3 Using Hybrid Learning Algorithm

Simulation.I

Fig.4.13 shows the array factor values in db for five elements at five angle of arrival and interferences using HLA. At aoa=20^0,30^0,40^0,50^0,60^0, we get array factor as -0.0711db, -0.0052db, -0.0051db, -0.0138db, -0.5023db respectively.At aoi=-20^0,-30^0,-40^0,-50^0,-60^0, we get array factor as -37.597db, -32.626db, -41.634db, -31.921db, -31.968db respectively.

Simulation.II

Fig.4.14 shows the array factor values in db for ten elements at five angle of arrival and interferences using HLA. At aoa=20^0,30^0,40^0,50^0,60^0, we get array factor as -0.0103db, -0.0368db, 0db, -0.0044db, 0db respectively.At aoi=-20^0,-30^0,-40^0,-50^0,-60^0, we get array factor as -27.157db, -30.967db, -28.201db, -41.064db, -38.605db respectively.

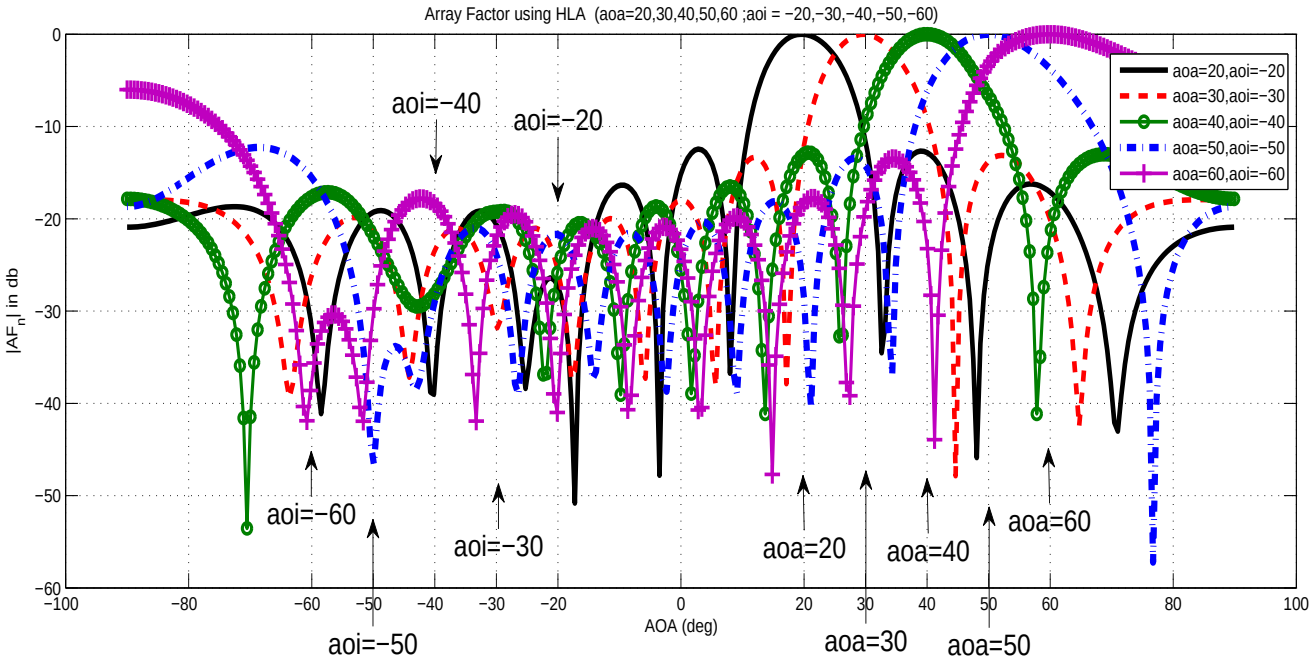

Figure 4.14: Array factor for five angle of arrivals using HLA (N=10)

4.4.4 Using HS algorithm

Simulation.I

Fig.4.15 shows the array factor values in db for five elements at five angle of arrival and interferences using HS. At aoa=20^0,30^0,40^0,50^0,60^0, we get array factor as -0.9592db, -0.5976db, -0.1489db, -0.2063db, -0.6392db respectively.At aoi=-20^0,-30^0,-40^0,-50^0,-60^0, we get array factor as -12.824db, -14.635db, -16.207db, -18.341db, -5.937db respectively.

Simulation.II

Fig.4.16 shows the array factor values in db for ten elements at five angle of arrival and interferences using HS. At aoa=20^0,30^0,40^0,50^0,60^0, we get array factor as -0.0446db, -0.5976db, -0.157db, -0.819db, -0.253db respectively.At aoi=-20^0,-30^0,-40^0,-50^0,-60^0, we get array factor as -14.810db, -19.635db, -14.674db, -12.348db, -26.266db respectively.

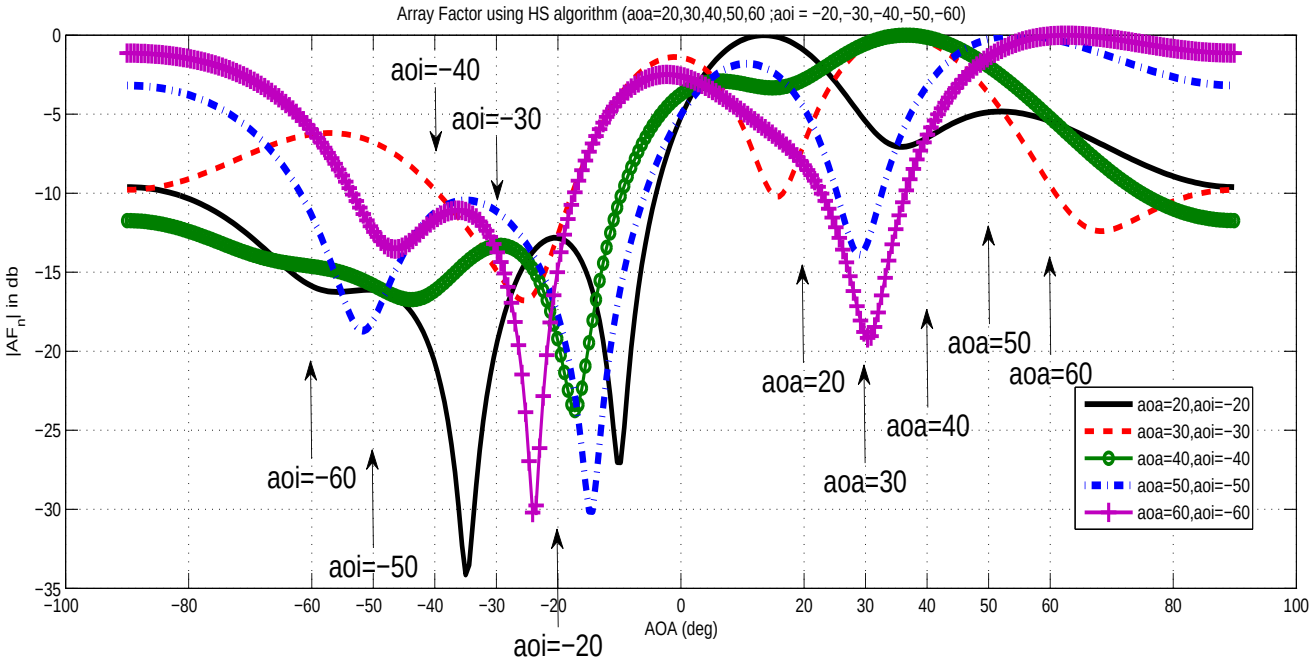

Figure 4.15: Array factor for five angle of arrivals using HS (N=5)

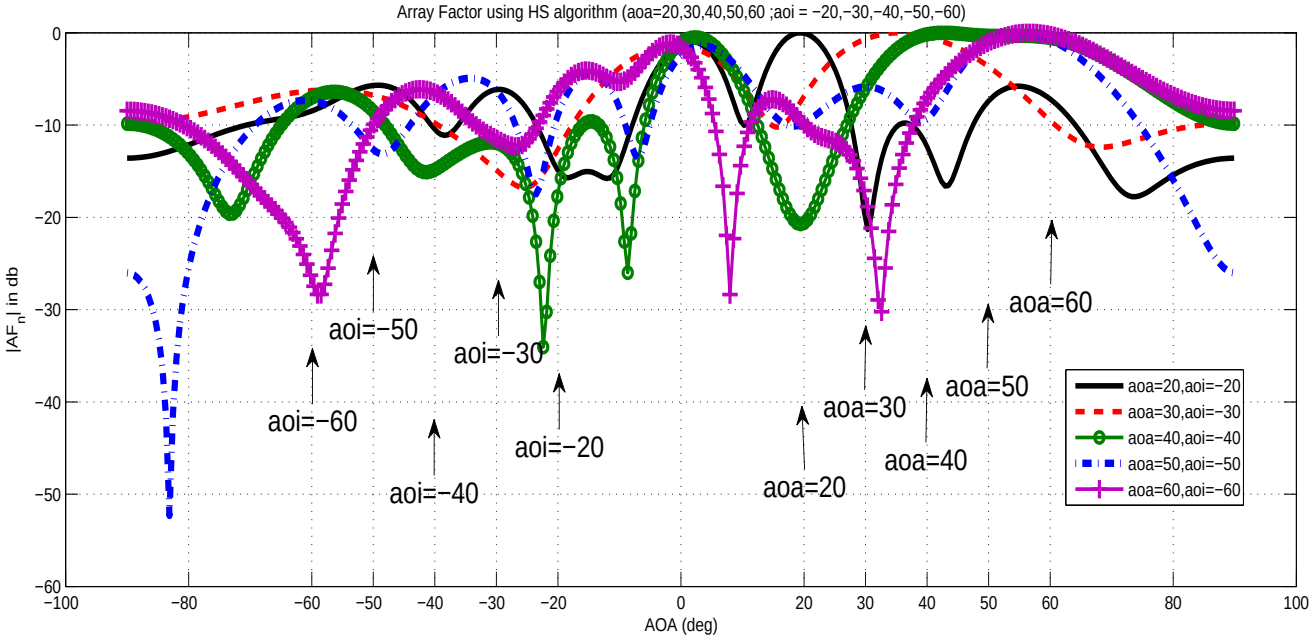

Figure 4.16: Array factor for five angle of arrivals using HS (N=10)

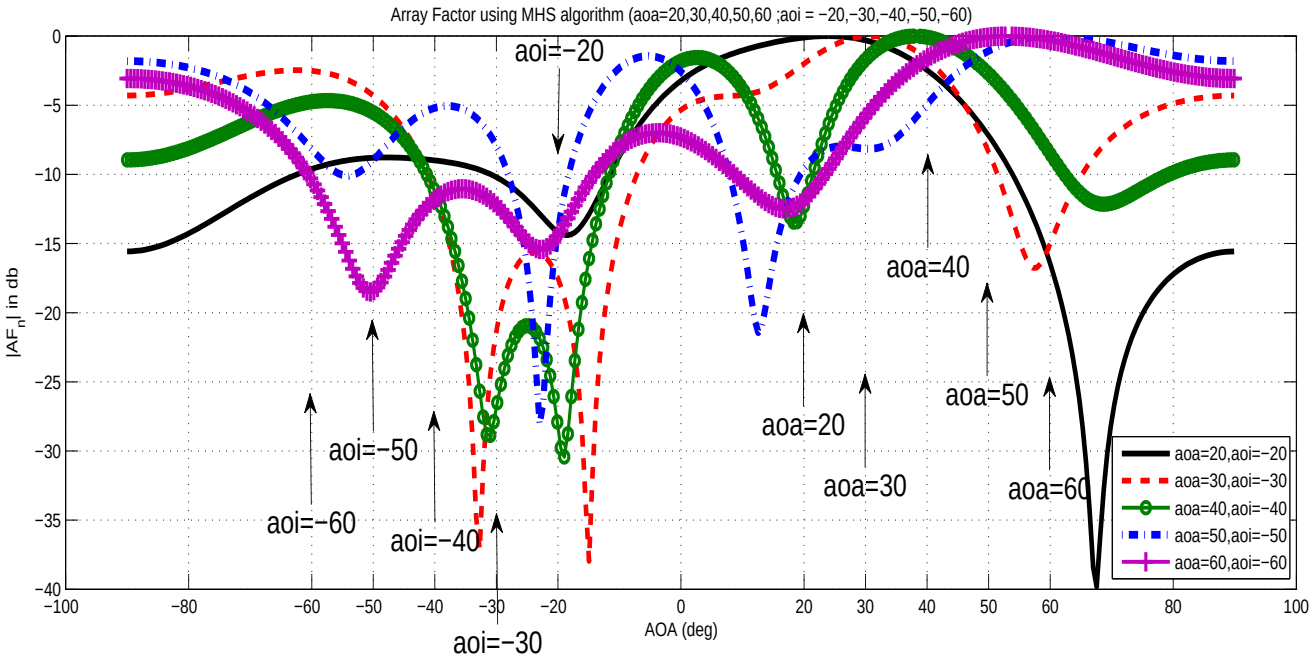

Figure 4.17: Array factor for five angle of arrivals using MHS (N=5)

4.4.5 Using MHS algorithm

Simulation.I

Fig.4.17 shows the array factor values in db for five elements at five angle of arrival and interferences using MHS. At aoa=20^0,30^0,40^0,50^0,60^0, we get array factor as -0.1012db, -0.0421db, -0.0823db, -0.9881db, -0.6189db respectively.At aoi=-20^0,-30^0,-40^0,-50^0,-60^0,we get array factor as -14.124db, -22.513db, -11.067db, -9.166db, -10.291db respectively.

Simulation.II

Fig.4.18 shows the array factor values in db for ten elements at five angle of arrival and interferences using MHS. At aoa=20^0,30^0,40^0,50^0,60^0, we get array factor as -0.0596db, -0.1477db, -1.916db, -0.0462db, -0.425db respectively.At aoi=-20^0,-30^0,-40^0,-50^0,-60^0, we get array factor as -14.368db, -22.513db, -8.976db, -12.312db, -19.943db respectively.

Table 4.3, 4.4, 4.5 and 4.6 shows array factor values in db for aoa=20^0,30^0,40^0,50^0,60^0 and aoi=-20^0,-30^0,-40^0,-50^0,-60^0, using different algorithms.Number of array

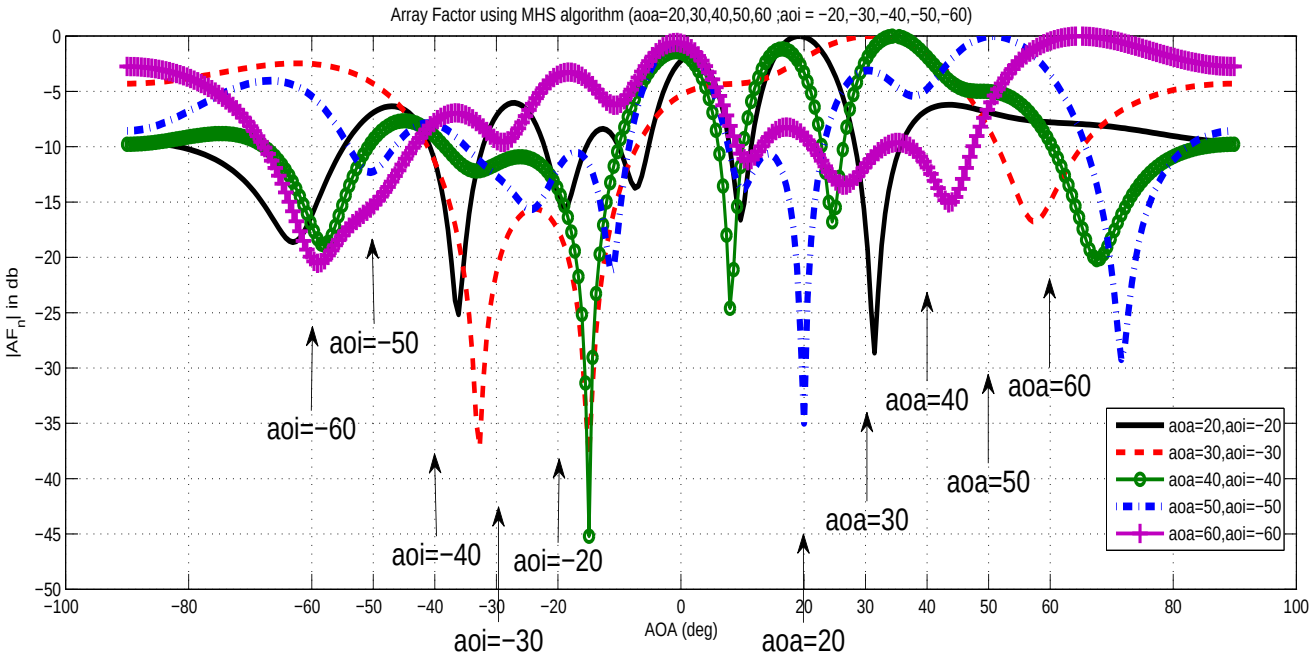

Figure 4.18: Array factor for five angle of arrivals using MHS (N=10)

elements are also varied.From the above simulations using five different algorithms we can conclude that Tuned LMS gives better array factor values for all five angle of arrival and interferences considered..

Algorithm	AOA (Array Factor in db)				
	20^0	30^0	40^0	50^0	60^0
Tuned LMS	-0.0115	-0.0007	-0.0019	-0.0093	-0.4515
RLS	-0.1117	-0.0149	-0.0641	-0.0613	-0.5471
HLA	-0.0711	-0.0052	-0.0051	-0.0138	-0.5023
HS	-0.9592	-0.5976	-0.1489	-0.2063	-0.6392
MHS	-0.1012	-0.0421	-0.0823	-0.9881	-0.6189

Table 4.3: Comparison between different algorithms for multiple angle of arrivals (N=5)

Algorithms	AOI (Array Factor in db)				
	-20^0	-30^0	-40^0	-50^0	-60^0
Tuned LMS	-53.556	-79.994	-50.936	-40.459	-42.905
RLS	-50.562	-61.147	-50.754	-39.309	-40.670
HLA	-37.597	-32.626	-41.634	-31.921	-31.968
HS	-12.824	-14.635	-16.207	-18.341	-5.937
MHS	-14.124	-22.513	-11.067	-9.166	-10.291

Table 4.4: Comparison between different algorithms for multiple angle of interferences (N=5)

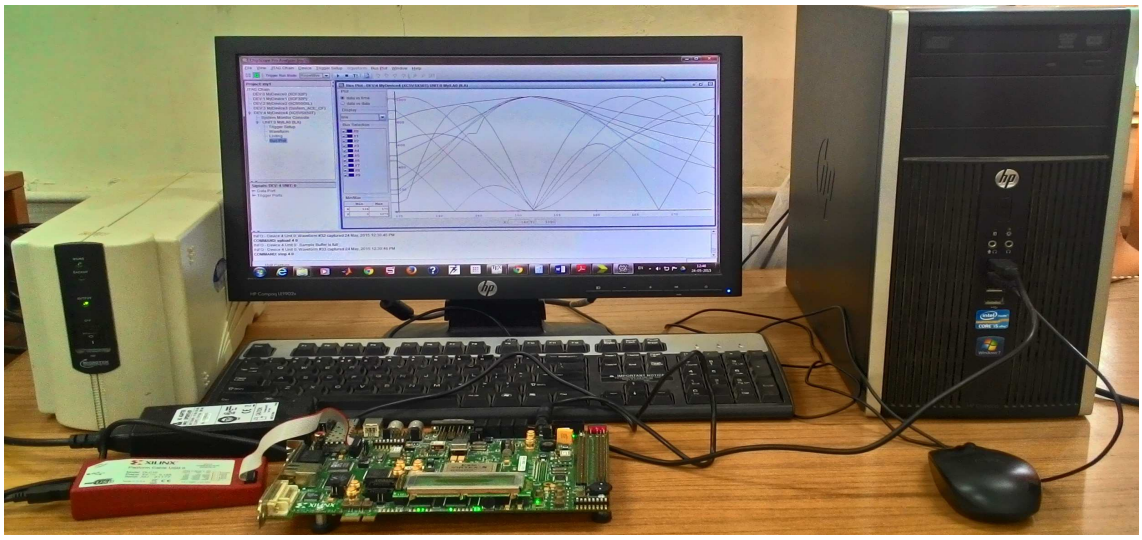

Figure 4.19: Experimental setup for FPGA device

Algorithm	AOA (Array Factor in db)				
	20^0	30^0	40^0	50^0	60^0
Tuned LMS	0	-0.0032	0	0	0
RLS	-0.0371	-0.048	0	0	-0.0043
HLA	-0.0103	-0.0368	0	-0.0044	0
HS	-0.0446	-0.5976	-0.157	-0.819	-0.253
MHS	-0.0596	-0.1477	-1.916	-0.0462	-0.425

Table 4.5: Comparison between different algorithms for multiple angle of arrivals (N=10)

Algorithm	AOI (Array Factor in db)				
	-20^0	-30^0	-40^0	-50^0	-60^0
Tuned LMS	-65.187	-40.208	-60.477	-45.822	-53.374
RLS	-60.304	-36.743	-57.758	-42.101	-52.753
HLA	-27.157	-30.967	-28.201	-41.064	-38.605
HS	-14.810	-19.635	-14.674	-12.348	-26.266
MHS	-14.368	-22.513	-8.976	-12.312	-19.943

Table 4.6: Comparison between different algorithms for multiple angle of interferences (N=10)

4.5 Experimental Setup

Fig.4.19 shows the experimental setup for fpga device. As shown in fig.4.19, Fpga device is connected to system using Xilinx platform USB cable.The programming file generated is used to configure the FPGA device.chipscope pro analyzer is utilized to observe the output waveforms in chipscope.

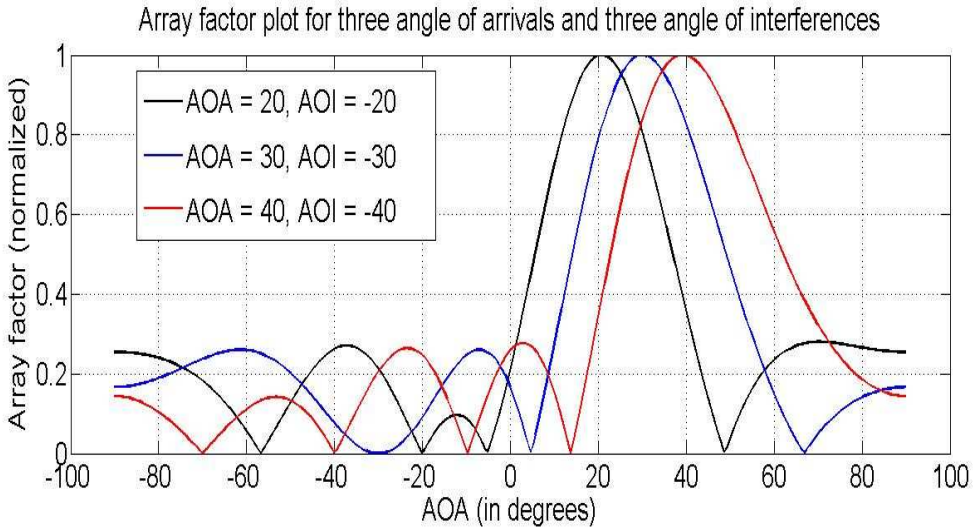

Figure 4.20: Array factor vs AOA for three AOA and AOI

4.6 Results captured on chipscope

VIRTEX-V FPGA is used to generate multiple beams.The results are taken through chipscope.Angle is varied and array factor values are calculated at different instant of time.Here,array factor values with respect to time is plotted using chipscope.

Result.I

Array factor for aoa=20^0,30^0,40^0 and aoi=-20^0,-30^0,-40^0 with respect to time is shown in Fig.4.21 and with respect to AOA is shown in Fig.4.20.

Result.II

Array factor for aoa=20^0,30^0,40^0,50^0 and aoi=-20^0,-30^0,-40^0,-50^0 with respect to time is shown in Fig.4.23 and with respect to AOA is shown in Fig.4.22.

Result.III

Array factor for aoa=20^0,30^0,40^0,50^0,60^0 and aoi=-20^0,-30^0,-40^0,-50^0,-60^0 with respect to time is shown in Fig.4.25 and with respect to AOA is shown in Fig.4.24.From fig.4.25 we observe that all angle of arrival and interferences occur at the same instant.

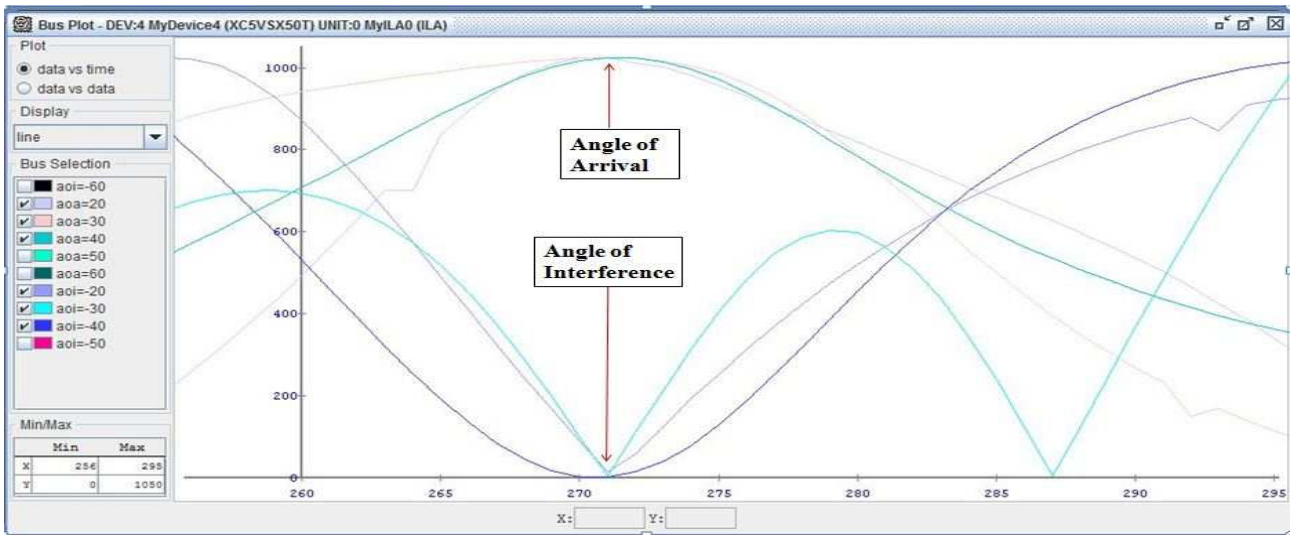

Figure 4.21: Chipscope output on VIRTEX-V FPGA for three angle of arrival and interferences

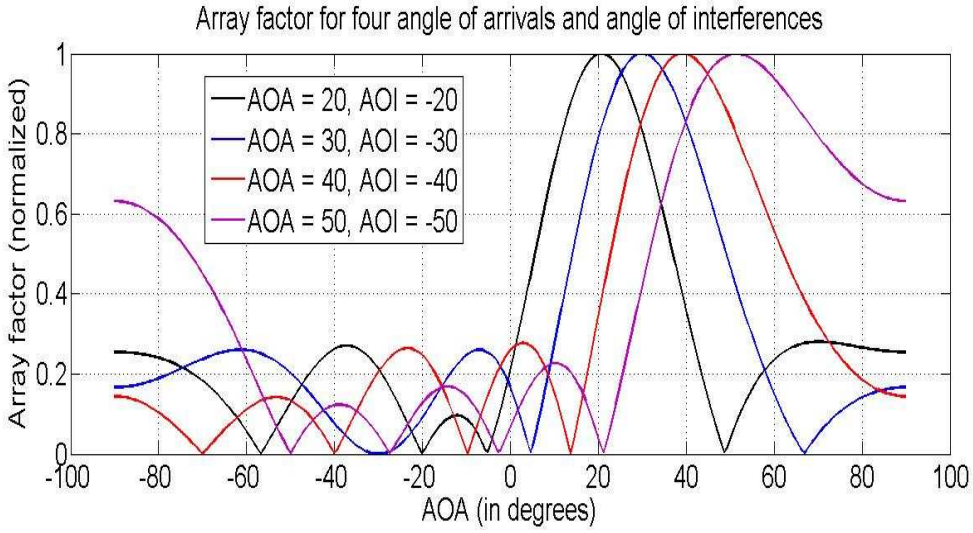

Figure 4.22: Array factor vs AOA for four AOA and AOI

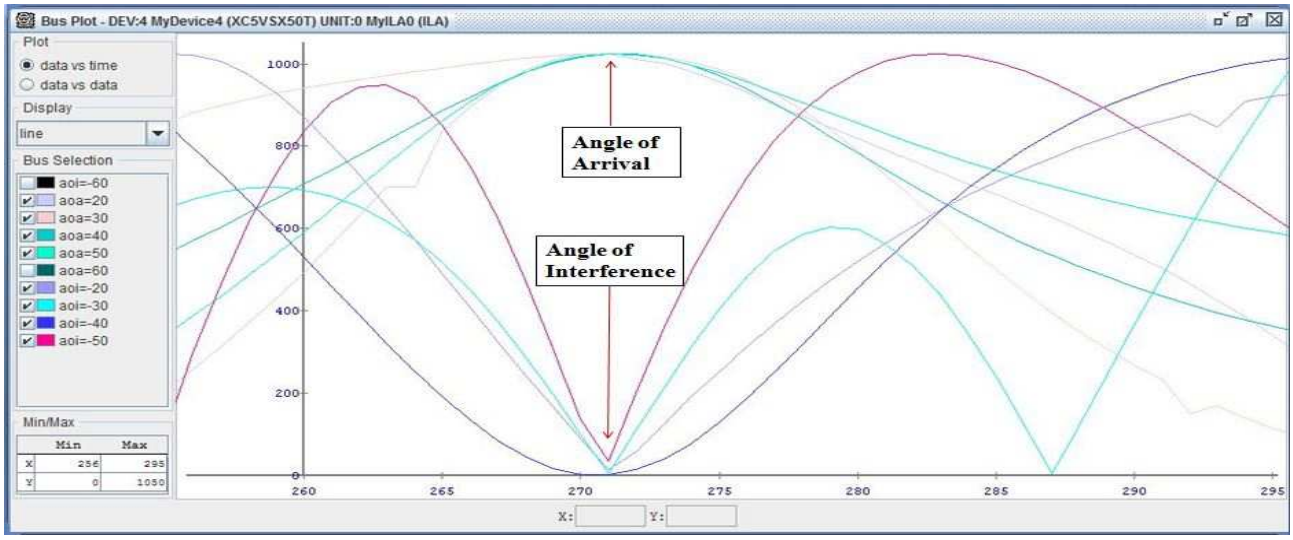

Figure 4.23: Chipscope output on VIRTEX-V FPGA for four angle of arrival and interferences

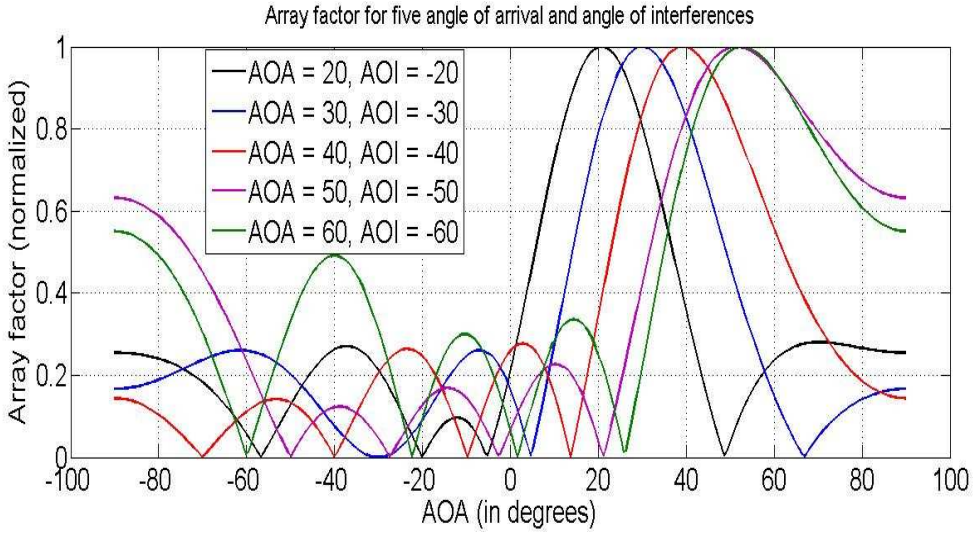

Figure 4.24: Array factor vs AOA for five AOA and AOI

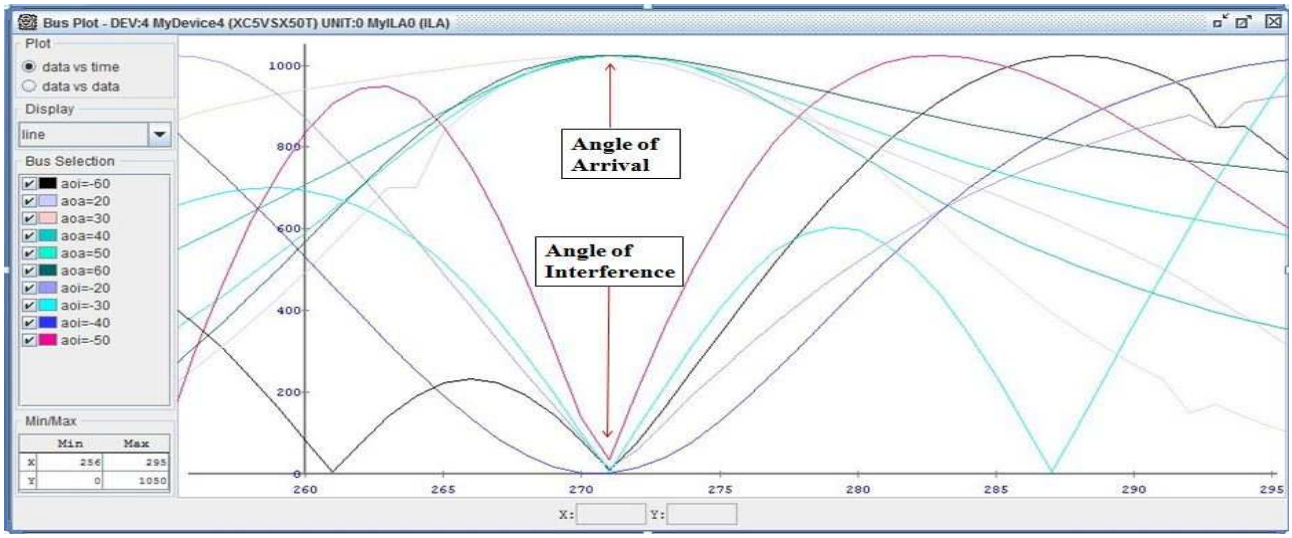

Figure 4.25: Chipscope output on VIRTEX-V FPGA for five angle of arrival and interferences

Chapter 5

CONCLUSION AND FUTURE SCOPE

Smart antenna systems are the systems in which the main beam is steered to the angle of direction and suppress the intensity level at the interference angles. In this work smart antenna system is designed and adapted using intelligent techniques for multiple arrivals and multiple interferences. The adaptive algorithms used are Tuned LMS,RLS,HLA,HS and MHS. The following conclusions can be drawn based on the simulation results of these techniques.

5.1 Conclusion

- In this thesis adaptive beam forming for linear geometry using different algorithms (Tuned LMS,RLS,HLA,HS and MHS) for a multiple objective function is considered.
- The evolutionary techniques used here are remodeled, establishing the analogy between the bio-inspired parameters and the antenna parameters.
- Different case studies including single and multiple arrival and interferences for multi-user concludes that Tuned LMS technique works better for the considered objectives. Using Tuned LMS technique both mean

square error and array factor value obtained are better then other algorithms.

- The performance of MHS is better than that of HS in terms of mean square error value.
- Another conclusion can be drawn based on algorithm parametric complexity which also portraits Tuned LMS as a better technique over other techniques.
- An attempt is made to improvise the technicality of HS and boost up its performance.
- A Neuro-Fuzzy approach has been used using hybrid learning algorithm (LSM and Backpropagation).

5.2 Limitations and Future Scope

- Through out the work the mutual coupling between the antenna array elements is neglected.
- Smart antenna array geometries are limited to only linear shapes.Different geometries can be considered for optimization.
- A neuro-fuzzy approach can be experimented to set initial parameters of MHS.
- Identical antenna elements are used in the array which can be changed.
- A lot of scope for the development of algorithms.Many different evolutionary algorithms can be experimented for better optimization.

Bibliography

[1] Magdalena salazar-Palma Tapan K.Sarkar, Michel C.Wicks and Robert J.Bonneau. *Smart Antennas*. Wiley-Interscience, 2003.

[2] Huilong Zhang, Xiurong Ma, Yunxiang Cheng, and Yuan Bai. Study on doa estimation algorithm for smart antenna of uniform circular. In *Computer Application and System Modeling (ICCASM), 2010 International Conference on*, volume 15, pages V15–627. IEEE, 2010.

[3] Ahmed El Zooghby. Smart antenna engineering, 2005 artech house. *INC., Norwood*.

[4] Zaharias D Zaharis, Christos Skeberis, and Thomas D Xenos. Improved antenna array adaptive beamforming with low side lobe level using a novel adaptive invasive weed optimization method. *Progress In Electromagnetics Research*, 124:137–150, 2012.

[5] Dhusar Kumar Mondal. Studies of different direction of arrival (doa) estimation algorithm for smart antenna in wireless communication.

[6] Vishal V Sawant and Mahesh Chavan. Performance of beamforming for smart antenna using traditional lms algorithm for various parameters.

[7] Revati Joshi and Ashwinikumar Dhande. Adaptive beamforming using lms algorithm.

[8] Miss Nayan B Shambharkar, Prakash M Mainkar, and GN Mulay. Implementation of rls beamforming algorithm for smart antenna system.

[9] Kimutai Victor Kimutai Victor Rop. *Parameter Optimization in Design of a Microstrip Patch Antenna Using Adaptive Neuro-Fuzzy Inference System Technique*. PhD thesis, 2014.

[10] Abhishek Rawat, RN Yadav, and SC Shrivastava. Neural network applications in smart antenna arrays: a review. *AEU-International Journal of Electronics and Communications*, 66(11):903–912, 2012.

[11] Zong Woo Geem, Joong Hoon Kim, and GV Loganathan. A new heuristic optimization algorithm: harmony search. *Simulation*, 76(2):60–68, 2001.

[12] Kang Seok Lee and Zong Woo Geem. A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice. *Computer methods in applied mechanics and engineering*, 194(36):3902–3933, 2005.

[13] Ren Diao and Qiang Shen. Two new approaches to feature selection with harmony search. In *Fuzzy Systems (FUZZ), 2010 IEEE International Conference on*, pages 1–7. IEEE, 2010.

[14] Swagatam Das, Arpan Mukhopadhyay, Anwit Roy, Ajith Abraham, and Bijaya K Panigrahi. Exploratory power of the harmony search algorithm: analysis and improvements for global numerical optimization. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 41(1):89–106, 2011.

[15] M Mahdavi, M Fesanghary, and E Damangir. An improved harmony search algorithm for solving optimization problems. *Applied mathematics and computation*, 188(2):1567–1579, 2007.

[16] Lucas M Pavelski, Carolina P Almeida, and Richard A Gonçalves. Harmony search for multi-objective optimization. In *Neural Networks (SBRN), 2012 Brazilian Symposium on*, pages 220–225. IEEE, 2012.

[17] D Govind Rao, NS Murthy, and A Vengadarajan. Design and implementation of digital beam former architecture for phased array radar.

[18] Robert S.Elliott. *ANTENNA THEORY AND DESIGN*. John Wiley and Sons, Ltd, 2005.

[19] Simon Haykin and Neural Network. A comprehensive foundation. *Neural Networks*, 2(2004), 2004.

[20] Bernard Widrow and Samuel D Stearns. Adaptive signal processing. *Englewood Cliffs, NJ, Prentice-Hall, Inc., 1985, 491 p.*, 1, 1985.

[21] Frank Gross. *Smart antennas for wireless communications*. McGraw-Hill Professional, 2005.

[22] Jyh-Shing Roger Jang and Chuen-Tsai Sun. *Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence*. Prentice-Hall, Inc., 1996.

[23] Ling Wang, Yin Xu, Yunfei Mao, and Minrui Fei. A discrete harmony search algorithm. In *Life System Modeling and Intelligent Computing*, pages 37–43. Springer, 2010.

Authors Biography

Manoj Govind Chaudhari was born to Mr. Govind Chaudhari and Mrs. Usha Chaudhari on 3rd March, 1992 at Mumbai, Maharashtra, India. He obtained Bachelor's degree in Electronics and Communication Engineering from Terna Engineering College,Navi Mumbai in 2013. He joined the Department of Electrical Engineering, National Institute of Technology, Rourkela in July 2013 as an Institute Research Scholar to pursue M.Tech degree in "Electronics System and Communication" specialization.